

django-mellon - Development #26143

ne pas faire de cache des métadonnées locales

05 septembre 2018 16:47 - Frédéric Péters

| | |
|--|---------------------------------------|
| Statut: Fermé | Début: 05 septembre 2018 |
| Priorité: Normal | Echéance: |
| Assigné à: | % réalisé: 0% |
| Catégorie: | Temps estimé: 0:00 heure |
| Version cible: | Planning: |
| Patch proposed: Non | |
| Description | |
| Ça ne semble pas apporter grand chose mais un bug, #26141 . | |
| Demandes liées: | |
| Lié à Hobo - Bug #26141: "race condition" sur la création d'un tenant puis la... | Fermé 05 septembre 2018 |
| Lié à django-mellon - Development #13881: ne pas utiliser les settings pour g... | Fermé 07 novembre 2016 |

Historique

#1 - 05 septembre 2018 16:50 - Frédéric Péters

- Fichier 0001-don-t-cache-local-metadata-anymore-26143.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

#2 - 05 septembre 2018 16:50 - Frédéric Péters

- Lié à Bug #26141: "race condition" sur la création d'un tenant puis la mise à dispo de la clé SAML ajouté

#3 - 05 septembre 2018 16:50 - Frédéric Péters

- Lié à Development #13881: ne pas utiliser les settings pour garder les métadonnées en cache ajouté

#4 - 05 septembre 2018 17:02 - Thomas Noël

Pourquoi pas, mais ça corrige pas vraiment le bogue, les metadata seront foireuses à un moment.

Je rapatrie ma question de [#26141](#) :

Je me demande dans quel cadre un SP SAML peut avoir des metadata sans clé publique ?

Auquel cas je préférerais que mellon réponde 404 quand app_settings.PUBLIC_KEYS est vide.

Quitte à gérer un possible "settings.METADATA_WITHOUT_PUBLIC_KEYS_IS_OK_GO_GO_GO == True" dans les settings (et 404 si cet attribut est absent)

#5 - 05 septembre 2018 17:11 - Frédéric Péters

- Statut changé de Solution proposée à Nouveau
- Assigné à Frédéric Péters supprimé
- Patch proposed changé de Oui à Non

#6 - 05 septembre 2018 18:06 - Emmanuel Cazenave

Thomas Noël a écrit :

Pourquoi pas, mais ça corrige pas vraiment le bogue, les metadata seront foireuses à un moment.

On passerait quand même de "foireuse de façon permanente (jusqu'à redémarrage dur serveur)" à "très brièvement foireuse", ça me semble être une avancée.

#7 - 05 septembre 2018 21:49 - Benjamin Dauvergne

- Statut changé de Nouveau à Solution validée

Thomas Noël a écrit :

Pourquoi pas, mais ça corrige pas vraiment le bogue, les metadata seront foireuses à un moment.

Je rapatrie ma question de [#26141](#) :

Je me demande dans quel cadre un SP SAML peut avoir des metadata sans clé publique ?

Quand il n'en a pas besoin :)

Auquel cas je préférerais que mellon réponde 404 quand `app_settings.PUBLIC_KEYS` est vide.

Quitte à gérer un possible `settings.METADATA_WITHOUT_PUBLIC_KEYS_IS_OK_GO_GO_GO == True` dans les settings (et 404 si cet attribut est absent)

On aurait juste besoin de pouvoir filer un template pour le répertoire du tenant à `create_tenant()`, histoire qu'il provisionne le contenu avant de renommer le répertoire.

#8 - 05 septembre 2018 21:59 - Frédéric Péters

Statut changé de Nouveau à Solution validée

Comme ce ticket parlait sur autre chose, j'ai déplacé le patch vers [#13881](#).

#9 - 05 septembre 2018 22:00 - Frédéric Péters

- Statut changé de Solution validée à En cours

Je laisse la discussion se terminer ici si elle doit l'être.

#10 - 06 septembre 2018 10:13 - Thomas Noël

(pas plus de discussion de ma part)

#11 - 06 septembre 2018 10:26 - Frédéric Péters

- Statut changé de En cours à Fermé

#12 - 06 septembre 2018 10:30 - Benjamin Dauvergne

Je serai pour fermer ce ticket et voir comment rendre l'initialisation d'un tenant plus atomique, dans cette partie:

```
tenant_dir = os.path.join(tenant_base, hostname)
if os.path.exists(tenant_dir):
    raise CommandError('tenant already exists')
tenant_dir_tmp = os.path.join(
    tenant_base, hostname + '__CREATION_IN_PROGRESS.invalid')
try:
    os.mkdir(tenant_dir_tmp, 0755)
except OSError as e:
    raise CommandError('cannot start tenant creation (%s)' % str(e))
# tenant creation in database
try:
    connection.set_schema_to_public()
    schema = TenantMiddleware.hostname2schema(hostname)
    tenant = get_tenant_model()(schema_name=schema, domain_url=hostname)
    if verbosity >= 1:
        print
        print self.style.NOTICE("=== Creating schema ") \
            + self.style.SQL_TABLE(tenant.schema_name)
    with transaction.atomic():
        tenant.create_schema(check_if_exists=True)
except Exception as e:
    os.rmdir(tenant_dir_tmp)
    raise CommandError('tenant creation failed (%s)' % str(e))
for folder in ('media', 'static', 'templates'):
    path = os.path.join(tenant_dir_tmp, folder)
    os.mkdir(path, 0755)
# activate the tenant by creating its directory
os.rename(tenant_dir_tmp, tenant_dir)
```

J'ai déjà signalé dans un autre ticket que la création du schéma devrait se faire dans une transaction (en cas d'échec sur l'initialisation du filesystem, on se retrouve avec un schéma en vrac, idem en cas de souci sur l'application des migrations, schéma incomplet). Actuellement c'est le répertoire qui fait foi quand à l'existence d'un tenant.

