Authentic 2 - Bug #15715

Sur une création de compte via API avec envoi de mail d'enregistrement, il faut définir un mot de passe par défaut

31 mars 2017 17:10 - Benjamin Dauvergne

Statut:	Fermé	**Début:**	31 mars 2017
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Non	**Planning:**	

Description

Sinon le formulaire de réinitialisation du mot de passe refusera la réinitialisation sous le motif que le compte ne dispose pas actuellement d'un mot de passe.

Historique

#2 - 31 mars 2017 17:11 - Benjamin Dauvergne

- Fichier 0001-api-set-random-on-user-creation-with-registration-ma.patch ajouté

- Patch proposed changé de Non à Oui

#3 - 01 avril 2017 14:03 - Frédéric Péters

Il y a des bouts dans setup.py et admin.py qui semblent d'ailleurs.

C'était galère de plutôt modifier la vue de réinitialisation de mot de passe pour ne pas bloquer sur l'absence de mot de passe ?

#4 - 01 avril 2017 14:13 - Benjamin Dauvergne

Frédéric Péters a écrit :

> Il y a des bouts dans setup.py et admin.py qui semblent d'ailleurs.
>
> C'était galère de plutôt modifier la vue de réinitialisation de mot de passe pour ne pas bloquer sur l'absence de mot de passe ?

Pour l'instant je me sers de l'absence de mot de passe pour dire que c'est un compte qui ne doit pas en avoir (typiquement un compte LDAP ou France Connect), Je veux bien rajouter un flag 'no_password_account' mais ça demanderait de réfléchir un peu plus.

#5 - 01 avril 2017 16:41 - Frédéric Péters

> Pour l'instant je me sers de l'absence de mot de passe pour dire que c'est un compte qui ne doit pas en avoir (typiquement un compte LDAP ou France Connect), Je veux bien rajouter un flag 'no_password_account' mais ça demanderait de réfléchir un peu plus.

Ok, restons sur l'approche du patch alors, mais pour être sûr que ça ne soit jamais bon, peut-être que le mot de passe pourrait être genre '\0' ?

#6 - 03 avril 2017 10:37 - Benjamin Dauvergne

Tout ce qui n'est pas une hash de mot de passe correct va faire que User.has_unusable_password() va retourner True et ça va bloquer l'écran de ré-initialisation, le plus simple c'est vraiment de mettre un truc random mais qui passe pour un vrai mot de passe (et si j'essaie de mettre un truc qui ressemble à un hash mais qui n'en est pas un j'ai peur que ça fasse des exceptions étranges quand on ne s'y attendra pas, par exemple sur une authentification HttpBasic dans django-rest-framework).

#7 - 03 avril 2017 10:42 - Benjamin Dauvergne

Benjamin Dauvergne a écrit :

> Tout ce qui n'est pas une hash de mot de passe correct va faire que User.has_unusable_password() va retourner True et ça va bloquer l'écran de ré-initialisation, le plus simple c'est vraiment de mettre un truc random mais qui passe pour un vrai mot de passe (et si j'essaie de mettre un truc qui ressemble à un hash mais qui n'en est pas un j'ai peur que ça fasse des exceptions étranges quand on ne s'y attendra pas, par exemple sur une authentification HttpBasic dans django-rest-framework).

Pour donner un peu de contexte si tu te souviens des versions précédentes de Django, avant un mot de passe inutilisable c'était seulement '!' mais ça a changé dans les versions récentes.

#8 - 03 avril 2017 10:43 - Frédéric Péters

Ok même si :

```
>>> user = User.objects.all()[0]
>>> user.set_password('\0')
>>> user.has_usable_password()
True
```

#9 - 03 avril 2017 11:10 - Benjamin Dauvergne

Bizarre en Django18 j'ai ce test qui passe et dit exactement le contraire:

```
+def test_login_no_password(db, app):
+    User = get_user_model()
+    user1 = User.objects.create(username='john.doe')
+    user1.password = ''
+    user1.save()
+    assert not user1.has_usable_password()
+
+    assert not authenticate(username=user1.username, password='coin')
+    assert not authenticate(username=user1.username, password='')
+
+    user1.password = '\0'
+    user1.save()
+    assert not user1.has_usable_password()
+
+    assert not authenticate(username=user1.username, password='coin')
+    assert not authenticate(username=user1.username, password='')
```

#10 - 03 avril 2017 11:11 - Frédéric Péters

Je fais .set_password().

#11 - 03 avril 2017 12:56 - Benjamin Dauvergne

Via l'API y a moyen d'envoyer '\0' comme mot de passe je pense en HTTPBasic (b64encode('login:\0')).

#12 - 03 avril 2017 12:56 - Benjamin Dauvergne

Et via le formulaire on doit pouvoir faire avec de l'urlencoding '%00'.

#13 - 03 avril 2017 13:14 - Frédéric Péters

C'était une crainte en ayant la flemme d'essayer; bref comme je disais plus haut, ok pour l'approche.

#14 - 17 novembre 2017 02:17 - Benjamin Dauvergne

- Statut changé de Nouveau à Résolu (à déployer)

- Patch proposed changé de Oui à Non

Corrigé par

```
commit f539e24e02d083a973cf0f3f34b7ccabfb476a6c
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Wed Jul 19 23:59:46 2017 +0200

    utils: reset user password in build_reset_password_url (fixes #18643)

    Two reasons:
    * if we ask to reset the user password, we need to invalidate the current password,
    * if we use the reset password view to set the password for a new user, we need to
      set a first password so that the user is not considered "passwordless", it's
      the case for creation through the backoffice and the API.
```

#15 - 06 décembre 2017 15:33 - Benjamin Dauvergne

- Statut changé de Résolu (à déployer) à Fermé

Fichiers

0001-api-set-random-on-user-creation-with-registration-ma.patch	1,81 ko	31 mars 2017	Benjamin Dauvergne