

## Authentic 2 - Development #68745

### rendre la force minimum de mot de passe requise configurable

06 septembre 2022 09:38 - Corentin Séchet

<b>Statut:</b>	Fermé	<b>Début:</b>	06 septembre 2022
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Corentin Séchet	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	Non
<b>Patch proposed:</b>	Oui		
<b>Description</b>			
Ajouter un setting en rapport sur les OU			

#### Révisions associées

##### Révision c393adcd - 07 octobre 2022 08:48 - Corentin Séchet

misc: make minimum password strength configurable in ous (#68745)

#### Historique

##### #1 - 07 septembre 2022 13:47 - Corentin Séchet

- Fichier 0001-misc-make-minimum-password-strength-configurable-in-.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

##### #2 - 15 septembre 2022 10:34 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

Le code est moins lisible, je me demande si ce ne serait pas plus propre de laisser `validate_password` dans `NewPasswordField` (dans une méthode clean pas dans `validators`, pour pouvoir lui passer un paramètre) et de passer explicitement `password_strength` au `Field` dans le `init` des `Form` plutôt que de jongler entre `Form.__init__`, `Form.__clean__` et `widget.min_strength`.

Il faudrait que la force du mot de passe suive le chemin d'abstraction `Form -> Field -> Widget`, au niveau `Form` on s'occupe de récupérer la force minimale sur l'utilisateur ou l'ou mais c'est tout (au niveau du `Field` `min_strength` doit être un `property`, pour réagir à la modification et modifier le widget en conséquence comme `ChoiceField.choices`).

##### #3 - 15 septembre 2022 11:20 - Benjamin Dauvergne

Et ce serait plutôt dans `Field.validate()` que `Field.clean()`.

##### #4 - 21 septembre 2022 14:18 - Corentin Séchet

- Fichier 0001-misc-make-minimum-password-strength-configurable-in-.patch ajouté
- Statut changé de En cours à Solution proposée

##### #5 - 28 septembre 2022 15:34 - Valentin Deniaud

Je trouve ce patch long et par conséquent difficile à relire, pour faciliter les choses on pourrait avoir d'abord un commit qui se borne à bouger du code (déplacement de `validate_password`, des tests) et un deuxième qui fasse les mods qui correspondent au titre du ticket.

Sur la forme,

- Je ne pense pas que `default=None` soit utile ?
- Pour l'ajout d'une `property` je trouve plus lisible un « `@property` » avant la définition de la méthode.
- `get_min_password_strength` ça donne envie d'en faire une méthode du modèle `User`.

##### #6 - 28 septembre 2022 16:15 - Benjamin Dauvergne

Valentin Deniaud a écrit :

- Je ne pense pas que `default=None` soit utile ?

Il n'est effectivement pas nécessaire mais personnellement je préfère quand la valeur par défaut est explicite même si c'est None; ça évite de réfléchir quand on survole ça.

- Pour l'ajout d'une property je trouve plus lisible un « @property » avant la définition de la méthode.

J'ai pas un avis super tranché là dessus, avec @property ce qui est chiant c'est que toutes les méthodes doivent s'appeler pareil quand on lit la doc de Python.

```
@property
def x...

@x.setter
def x...

@x.deleter
def x...
```

Mais on peut faire un truc comme ça :

```
class A:
    @property
    def x(self):
        print('get x')

    def set_x(self, u):
        print('set x to', u)

x = x.setter(set_x)
```

- get\_min\_password\_strength ça donne envie d'en faire une méthode du modèle User.

Pas vraiment, c'est une propriété de l'OU ou globale, pas des utilisateurs, ce qui est bizarre c'est de passer par l'utilisateur, ça laisse penser que ça dépend de lui (de l'objet User) alors que non.

Et au niveau de RegistrationForm ça dépend du comportement un peu particulier qu'une instance User est initialisée avec son ou, sans être sauvegardée avant de construire l'objet ModelForm, je trouverai plus clair de voir ici référencer l'ou par défaut. Clairement RegistrationCompletionView est tout moche, je ne pense pas qu'on puisse tout nettoyer dans ce ticket. get\_min\_password\_strength() peut rester.

---

De mon côté je remarque un test test\_registraton\_completion\_form qui s'est retrouvé bizarrement dans test\_password\_reset, il devrait être dans test\_registraton, et il faudrait probablement aussi un test passant par les vues pour tester complètement. (avec ça je validerai)

#### #7 - 06 octobre 2022 12:31 - Corentin Séchet

- Fichier 0001-misc-make-minimum-password-strength-configurable-in-.patch ajouté

#### #8 - 06 octobre 2022 23:59 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

#### #9 - 07 octobre 2022 08:59 - Corentin Séchet

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

```
commit c393adcdd59564254d71847a14e1a1780eee552e
Author: Corentin Séchet <csechet@entrouvert.com>
Date: Tue Sep 6 18:02:12 2022 +0200
```

```
misc: make minimum password strength configurable in ous (#68745)
```

#### #10 - 07 octobre 2022 10:14 - Transition automatique

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

#### #11 - 11 décembre 2022 04:42 - Transition automatique

Automatic expiration

### Fichiers

---

0001-misc-make-minimum-password-strength-configurable-in-.patch 19,8 ko 07 septembre 2022

Corentin Séchet

0001-misc-make-minimum-password-strength-configurable-in-.patch 23,2 ko 21 septembre 2022  
0001-misc-make-minimum-password-strength-configurable-in-.patch 23,9 ko 06 octobre 2022

Corentin Séchet  
Corentin Séchet