# Authentic 2 - Development #33515

## Multi-facteurs : mécanique django_rbac et utilisation dans le manager

28 mai 2019 17:18 - Valentin Deniaud

| | | | |
|---|---|---|---|
| **Statut:** | Fermé | **Début:** | 28 mai 2019 |
| **Priorité:** | Normal | **Echéance:** | |
| **Assigné à:** | | **% réalisé:** | 0% |
| **Catégorie:** | | **Temps estimé:** | 0:00 heure |
| **Version cible:** | | | |
| **Patch proposed:** | Oui | **Planning:** | Non |

**Description**

L'interface qu'on expose via django_rbac étant conditionnée par l'interface qu'on a envie de manipuler dans le manager, je mets les deux ensembles.

Notes informatives sur la partie rbac : on associe un niveau d'authentification à un rôle. Pour un utilisateur doté lui aussi d'un niveau d'authentification, on veut permettre de vérifier s'il a accès ou pas à certaines permissions en excluant celles données par des rôles dont le niveau est trop élevé.
Plus précisément, à la question « est-ce que l'utilisateur a ces permissions ? », on doit pouvoir répondre oui ou non comme avant, mais aussi « non, mais il pourrait les avoir avec un niveau supérieur ». Ou alors obliger à poser à chaque fois deux questions, mais ça ferait du code moche et ce n'est pas l'approche retenue ici.
Le niveau d'un utilisateur étant stocké dans sa session django (et pas dans un attribut de User facilement accessible), on permet aux méthodes de la famille User.has_perm de recevoir un nouveau kwarg auth_level, qui doit indiquer le niveau de l'utilisateur courant. Si il n'est pas présent, tout marche comme d'habitude.
Si il est présent, on vérifie d'abord si l'utilisateur a les permissions en regardant celles données par les rôles de l'utilisateur ayant un niveau inférieur ou égal au sien. Si c'est bon, on retourne. Sinon, on vérifie à nouveau, sans filtrer les rôles. Si c'est pas bon, on retourne. Si en revanche, cette fois-ci ça passe, on lève une exception, qui informe que ça pourrait passer avec un niveau plus élevé.

Notes informatives sur la partie manager : le cas de base est très simple à gérer, on modifie légèrement le PermissionMixin. En revanche, il y a plein de cas particuliers qui font que c'est pas si joli au final. Notamment les formulaires qui s'ouvrent dans des popups : si une montée de niveau est déclenchée à ce moment là, la redirection vers la page de login va (mal) s'afficher dans la popup.

Plus d'infos dans les messages de commit :)

### Historique

**#1 - 28 mai 2019 17:23 - Valentin Deniaud**

*- Patch proposed changé de Non à Oui*

**#2 - 28 mai 2019 17:24 - Valentin Deniaud**

*- Fichier 0007-manager-use-could_-action-instead-of-can_-in-templat.patch ajouté*

*- Fichier 0001-django_rbac-add-authentication-level-field-to-Role-m.patch ajouté*

*- Fichier 0005-manager-differentiate-perm-granted-while-ignoring-au.patch ajouté*

*- Fichier 0002-django_rbac-allow-filtering-user-roles-by-auth-level.patch ajouté*

*- Fichier 0003-django_rbac-add-auth_level-arg-to-permission-methods.patch ajouté*

*- Fichier 0006-manager-handle-special-cases-of-access-control-33515.patch ajouté*

*- Fichier 0008-manager-disable-popup-display-on-insufficient-auth-l.patch ajouté*

*- Fichier 0004-manager-check-authentication-level-in-PermissionMixi.patch ajouté*

*- Statut changé de Nouveau à Solution proposée*

**#3 - 05 juin 2019 13:27 - Valentin Deniaud**

*- Fichier 0001-django_rbac-add-authentication-level-field-to-Role-m.patch supprimé*

**#4 - 05 juin 2019 13:28 - Valentin Deniaud**

*- Fichier 0001-django_rbac-add-authentication-level-field-to-Role-m.patch ajouté*

Ni vu ni connu, un tuple qui se transforme en liste pour la compatibilité 1.8.

#### #5 - 05 juin 2019 14:09 - Emmanuel Cazenave

Benjamin dira mais notre mouvement global est à l'abandon de 1.8

#### #6 - 12 juin 2019 14:51 - Valentin Deniaud

*- Fichier 0006-manager-handle-special-cases-of-access-control-33515.patch ajouté*

Correction d'un petit bug, mais il faudra sûrement réfléchir à améliorer les quelques bizarreries dans la gestion des permissions au lieu de construire aveuglément par dessus (genre fixer #20513).
Aussi il reste un bug : les boutons de suppression dans les listes se comportent mal si ils déclenchent une montée de niveau. Il y a plein de manières d'y remédier, je laisse ça pour plus tard.

#### #7 - 10 février 2020 10:28 - Valentin Deniaud

*- Statut changé de Solution proposée à Nouveau*

*- Assigné à Valentin Deniaud supprimé*

#### #8 - 22 novembre 2021 11:24 - Valentin Deniaud

*- Statut changé de Nouveau à Fermé*

### Fichiers

| | | | |
|---|---|---|---|
| 0007-manager-use-could_-action-instead-of-can_-in-templat.patch | 12,8 ko | 28 mai 2019 | Valentin Deniaud |
| 0005-manager-differentiate-perm-granted-while-ignoring-au.patch | 4 ko | 28 mai 2019 | Valentin Deniaud |
| 0002-django_rbac-allow-filtering-user-roles-by-auth-level.patch | 1,61 ko | 28 mai 2019 | Valentin Deniaud |
| 0003-django_rbac-add-auth_level-arg-to-permission-methods.patch | 7,29 ko | 28 mai 2019 | Valentin Deniaud |
| 0006-manager-handle-special-cases-of-access-control-33515.patch | 3,32 ko | 28 mai 2019 | Valentin Deniaud |
| 0008-manager-disable-popup-display-on-insufficient-auth-l.patch | 7,8 ko | 28 mai 2019 | Valentin Deniaud |
| 0004-manager-check-authentication-level-in-PermissionMixi.patch | 3,09 ko | 28 mai 2019 | Valentin Deniaud |
| 0001-django_rbac-add-authentication-level-field-to-Role-m.patch | 3,29 ko | 05 juin 2019 | Valentin Deniaud |
| 0006-manager-handle-special-cases-of-access-control-33515.patch | 5,46 ko | 12 juin 2019 | Valentin Deniaud |