

w.c.s. - Bug #76354

accès api, un dispatch dans le workflow enlève le droit en lecture

06 avril 2023 17:10 - Valentin Deniaud

Statut:	Fermé	Début:	06 avril 2023
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Non		
Description			
Pour un formulaire donné, si on affecte un rôle A à une fonction B, alors un client d'API qui dispose du rôle A donne accès aux demandes via l'API.			
Si au cours du workflow il y a une liaison fonction/rôle qui affecte un rôle C à la fonction B, l'accès en lecture via l'API est perdu, il ne faudrait pas (pour correspondre au comportement côté consultation).			
Demandes liées:			
Lié à w.c.s. - Autre #76339: Accès via client d'API, toutes les fonctions n'o...		Rejeté	06 avril 2023
Lié à w.c.s. - Development #76682: ne pas inclure la fonction attribuée au ni...		Fermé	16 avril 2023

Historique

#1 - 06 avril 2023 17:10 - Valentin Deniaud

- Lié à Autre #76339: Accès via client d'API, toutes les fonctions n'ont pas les droits en lecture ? ajouté

#2 - 12 avril 2023 11:29 - Valentin Deniaud

- Fichier 0001-wip-dispatch-default-role-permission.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Avant de s'intéresser à l'API, il faudrait déjà que ça fonctionne comme attendu en backoffice.

Frédéric Péters a écrit ([#76339#note-3](#)) :

Quand un rôle est posé par défaut, l'accès en lecture devrait persister, même si une action de dispatch modifie l'assignation de la fonction. (c'est sûr le cas pour la consultation, il manque peut-être quelque chose côté API).

Je n'ai pas réussi à obtenir ce comportement en cliquant sur mon instance de dev, et là j'ai écrit un test qui montre que ça ne fonctionne pas comme ça. Il y a un truc que j'ai compris de travers ?

#3 - 16 avril 2023 13:10 - Frédéric Péters

- Statut changé de Solution proposée à En cours

- Patch proposed changé de Oui à Non

#4 - 16 avril 2023 13:51 - Frédéric Péters

- Statut changé de En cours à Fermé

De ce que je relis, aujourd'hui ce comportement existe uniquement pour le filtre "fonction de l'utilisateur connecté" (qui exploite formdata.workflow_merged_roles_dict), très historiquement (et assez long à retrouver) c'était le cas, jusque fc57c00f (2013, pas de ticket) où le raccourci de se baser sur le paramétrage fait au niveau du formdef est retiré pour tout le temps faire confiance au formdata, ça modifie le formdef.is_user_allowed_read ainsi :

```
if self.is_of_concern_for_user(user):
-     return True
+     if not formdata:
+         return True
```

ça fait donc 10 ans qu'on vit avec ça, on ne va pas changer, il y aurait par contre à corriger le workflow_merged_roles_dict pour correspondre, je ferai

un ticket dédié.

#5 - 16 avril 2023 13:53 - Frédéric Péters

- Lié à Development #76682: ne pas inclure la fonction attribuée au niveau du formdef dans le filtrage sur les fonctions de l'utilisateur ajouté

Fichiers

0001-wip-dispatch-default-role-permission.patch	1,76 ko	12 avril 2023	Valentin Deniaud
---	---------	---------------	------------------