

Authentic 2 - Development #50889

API pour obtenir la liste des membres d'un rôle

04 février 2021 18:07 - Nicolas Roche

Statut:	Fermé	Début:	04 février 2021
Priorité:	Normal	Echéance:	
Assigné à:	Nicolas Roche	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Le travaille est déjà en parti fait dans #25645 , mais dans l'autre sens.			

Révisions associées

Révision 852655fb - 30 mars 2021 21:47 - Nicolas Roche

api: upgrade change_role permission to manage_members (#50889)

Révision ed01a4e8 - 30 mars 2021 21:47 - Nicolas Roche

api: add list and retrieve role member(s) api (#50889)

Révision b21d5a56 - 30 mars 2021 21:47 - Nicolas Roche

tests: copy/paste users tests to member users api (#50889)

Historique

#3 - 17 février 2021 12:23 - Nicolas Roche

- Assigné à mis à Nicolas Roche

#5 - 17 février 2021 19:05 - Nicolas Roche

- Fichier 0001-api-add-role-members-on-list-roles-api-50889.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

J'ai choisi de retourner deux liste : une pour les membres directs et une pour tous les membres (directs + hérités).
Les utilisateurs renseignés dans ces deux listes contiennent les champs username , email , first_name , last_name et uuid .

Pour tester :

```
curl https://LOGIN:PASSWD@authentic.dev.publik.love/api/roles/
```

#6 - 17 février 2021 23:47 - Thomas Noël

J'ai l'impression que tu charges une API déjà existante, et que ça pourrait l'alourdir considérablement ?

Aussi je ne vois pas bien comment deux valeurs identiques peuvent renvoyer des listes différentes :

```
members = AllMembersListingField(many=True, read_only=True)
all_members = AllMembersListingField(many=True, read_only=True)
```

#7 - 18 février 2021 09:54 - Nicolas Roche

- Fichier 0001-api-add-role-members-on-list-roles-api-50889.patch ajouté

- Statut changé de Solution proposée à Information nécessaire

(patch qui corrige juste le test)

je ne vois pas bien comment deux valeurs identiques peuvent renvoyer des listes différentes

cf <https://www.django-rest-framework.org/api-guide/relations/#custom-relational-fields>

moi aussi je suis bluffé par la magie de DRF, qui j'imagine fait de l'introspection sur le nom de la variable.

tu charges une API déjà existante,

Oui, dans l'idée de continuer à utiliser DRF. Par exemple cette URL retourne également les membres du groupe :

```
curl https://LOGIN:PASSWD@authentic.dev.publik.love/api/roles/UUID_DU_ROLE
```

et que ça pourrait l'alourdir considérablement ?

Oui, est-ce à dire que ce que l'on souhaite ici c'est plutôt d'ajouter 2 nouvelles API roles/X/members et roles/X/all_members ?

#8 - 18 février 2021 10:04 - Benjamin Dauvergne

Je vote pour un sous endpoint `/api/roles/<uuid>/members/`, implémenté via UsersAPI pour que ça se comporte comme l'API des utilisateurs avec le même retour, la même recherche et pagination et dans un premier temps on uniquement le queryset `all_members`, pas besoin de faire plus sioux (dans la plupart des cas on s'en fout, on veut tout le monde, l'héritage pour l'instant c'est un détail technique interne à A2, toutes les briques l'ignore).

PS: merci de poser un plan, même vague avant d'écrire du code, c'est dommage de coder pour rien.

PS2: on ne peut pas vraiment reprendre UsersAPI tel quel, il faut ignorer tout ce qui permet de créer/modifier un utilisateur, donc juste les endpoints `list` et `retrieve`.

#9 - 22 février 2021 10:22 - Emmanuel Cazenave

Benjamin Dauvergne a écrit :

et dans un premier temps on uniquement le queryset `all_members`, pas besoin de faire plus sioux (dans la plupart des cas on s'en fout, on veut tout le monde, l'héritage pour l'instant c'est un détail technique interne à A2, toutes les briques l'ignore).

Pour le cas d'usage Lorrain qui a motivé la création de ce ticket je pense que ça a une importance de pouvoir distinguer membre direct/héritage, parce qu'ils vont donner/enlever des rôles aux utilisateurs via l'API.

#10 - 22 février 2021 11:36 - Benjamin Dauvergne

Ok, alors faire un endpoint `members` et un paramètre `nested=true/false`, par défaut à `false`.

#11 - 23 février 2021 08:21 - Nicolas Roche

- Fichier `0003-tests-copy-paste-users-tests-to-member-users-api-508.patch` ajouté

- Fichier `0002-api-add-a-list-role-members-api-50889.patch` ajouté

- Fichier `0001-misc-correct-indentation-50889.patch` ajouté

- Statut changé de *Information nécessaire* à *Solution proposée*

Voilà.

on ne peut pas vraiment reprendre UsersAPI tel quel, il faut ignorer tout ce qui permet de créer/modifier un utilisateur, donc juste les endpoints `list` et `retrieve`.

cf <https://www.django-rest-framework.org/api-guide/viewsets/> :

```
roles_members = RolesMembersAPI.as_view({'get': 'list'})
```

(J'ai l'impression que `retrieve` devra nécessairement faire l'objet d'un nouveau endpoint `/api/roles/<role_uuid>/members/<user_uuid>`, et que l'on s'éloigne alors du sujet de ce ticket qui vise à obtenir une liste.)

#12 - 24 février 2021 22:32 - Benjamin Dauvergne

Nicolas Roche a écrit :

(J'ai l'impression que `retrieve` devra nécessairement faire l'objet d'un nouveau endpoint `/api/roles/<role_uuid>/members/<user_uuid>`, et que l'on s'éloigne alors du sujet de ce ticket qui vise à obtenir une liste.)

De fait on a pas actuellement de quoi faire varier la liste des champs renvoyer par UsersAPI (ou alors peut-être que ça existe et que je ne connais pas bien `django-rest-framework`, dans un sens ça m'arrangerait). Mais oui idéalement on aurait quelque chose de minimum et peu coûteux (i.e. sans

les attributs étendus dans /members/ et l'utilisateur complet dans /members/<uuid>/), pour l'instant je dirais de faire au mieux avec ce qu'on a.

#13 - 28 février 2021 15:59 - Nicolas Roche

- Fichier 0003-tests-copy-paste-users-tests-to-member-users-api-508.patch ajouté
- Fichier 0002-api-add-a-list-role-members-api-50889.patch ajouté
- Fichier 0001-misc-correct-indentation-50889.patch ajouté

De ma compréhension de DRF, on n'a pas de quoi gérer les routers complexes :

<https://www.django-rest-framework.org/api-guide/routers/#drf-nested-routers>

Et donc je n'ai pas réussi à factoriser pour ne pas avoir à introduire des nouvelle URLs comme pour listers les membres, et à avoir un unique routeur comme c'est fait pour UsersAPI.

Ce patch ajoute 2 endpoints :

- GET roles/UUID/members/{?nested=true}
- GET roles/UUID/members/UUID/{?nested=true}

Les retours sont les mêmes que ceux obtenus avec UsersAPI :

- GET users/ :

```
{
  "next" : null,
  "previous" : null,
  "results" : [
    ...
  ]
}
```

ou

```
{"next":null,"previous":null,"results":[]}
```

- GET user/UUID/

```
{
  ...
  "first_name" : "Nicolas",
  "first_name_verified" : false,
  "id" : 2,
  "is_active" : true,
  ...
}
```

ou

```
{"result":0,"errors":{"detail":"Pas trouvé."}}
```

quelque chose de minimum et peu coûteux (i.e. sans les attributs étendus dans /members/ et l'utilisateur complet dans /members/<uuid>/),

J'ai simplifié le queryset utilisé en pensant que ça suffirait mais en fait non, ça ne change pas le rendu : par exemple la date de naissance est toujours retournée que avec GET users/.
(sur ce point je bloque)

#15 - 19 mars 2021 20:49 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

Nicolas Roche a écrit :

De ma compréhension de DRF, on n'a pas de quoi gérer les routers complexes :

<https://www.django-rest-framework.org/api-guide/routers/#drf-nested-routers>

Et donc je n'ai pas réussi à factoriser pour ne pas avoir à introduire des nouvelle URLs comme pour listers les membres, et à avoir un unique routeur comme c'est fait pour UsersAPI.

Ouaip je comprends comme toi. Ça aurait été plus joli d'introduire un viewset pour /api/roles/.../members/ mais ce sera pour une autre fois.

```
RolesMembersAPI.get_queryset(self)
```

C'est un peu moche, ici on peut juste utiliser directement `role.all_members()`.

```
qs = qs.filter(deleted__isnull=True)
```

`deleted` n'existe plus.

```
perm = 'a2_rbac.change_role'
```

Ça c'est un bug qui traîne, la permission pour la gestion des membres s'appelle `'manage_members'` maintenant.

Je serais plutôt pour que les checks restent simples et soient déplacés dans chaque méthode `get/post/delete` plutôt que de jouer avec `request.method`, 2 fois.

#16 - 22 mars 2021 12:19 - Nicolas Roche

- Fichier `0004-tests-copy-paste-users-tests-to-member-users-api-508.patch` ajouté
- Fichier `0003-api-add-list-and-retrieve-role-member-s-api-50889.patch` ajouté
- Fichier `0002-api-upgrade-change_role-permission-to-manage_members.patch` ajouté
- Fichier `0001-misc-correct-indentation-50889.patch` ajouté
- Statut changé de *En cours* à *Solution proposée*

Remarques prises en compte.

```
perm = 'a2_rbac.change_role'
```

traité dans 0002 (qui vient s'ajouter)

```
RolesMembersAPI.get_queryset(self)  
C'est un peu moche, ...
```

Oui, c'est tordu de rentrer dans la classe `RolesMembersAPI` par ici,

```
... ici on peut juste utiliser directement role.all_members().
```

mais je rajoute le `'if'` sur `'nested'` pour que sur `GET roles/UUID/members/UUID/` on ne retourne pas un utilisateur uniquement membre indirect du groupe.

#17 - 30 mars 2021 11:20 - Benjamin Dauvergne

- Statut changé de *Solution proposée* à *En cours*

Nicolas Roche a écrit :

Remarques prises en compte.

```
perm = 'a2_rbac.change_role'
```

traité dans 0002 (qui vient s'ajouter)

Oui mais non la gestion des rôles eux même (nom, slug, etc..) c'est bien `change_role`, la permission ne doit changer que dans `RoleMembershipAPI`.

```
... ici on peut juste utiliser directement role.all_members().
```

mais je rajoute le `'if'` sur `'nested'` pour que sur `GET roles/UUID/members/UUID/` on ne retourne pas un utilisateur uniquement membre indirect du groupe.

À voir si c'est une bonne idée mais ok.

#18 - 30 mars 2021 13:16 - Nicolas Roche

- Fichier 0002-api-upgrade-change_role-permission-to-manage_members.patch ajouté
- Statut changé de En cours à Solution proposée

Oui mais non la gestion des rôles eux même (nom, slug, etc..) c'est bien change_role, la permission ne doit changer que dans RoleMembershipAPI.

Merci pour ta vigilance.

#19 - 30 mars 2021 16:05 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

#20 - 30 mars 2021 21:49 - Nicolas Roche

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

```
commit b21d5a56b15d1fdf615919fbb93b614feac92193
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Mon Feb 22 23:13:38 2021 +0100
```

```
tests: copy/paste users tests to member users api (#50889)
```

```
commit ed01a4e8c34653f74217ca42cdb4bd68b45b1432
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Mon Mar 22 10:15:15 2021 +0100
```

```
api: add list and retrieve role member(s) api (#50889)
```

```
commit 852655fb95af9850a53b4430169719f4de3960d5
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Mon Mar 22 11:30:53 2021 +0100
```

```
api: upgrade change_role permission to manage_members (#50889)
```

#21 - 01 avril 2021 09:16 - Frédéric Péters

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

Fichiers

Fichier	Taille	Date	Auteur
0001-api-add-role-members-on-list-roles-api-50889.patch	5,63 ko	17 février 2021	Nicolas Roche
0001-api-add-role-members-on-list-roles-api-50889.patch	5,66 ko	18 février 2021	Nicolas Roche
0002-api-add-a-list-role-members-api-50889.patch	6,94 ko	23 février 2021	Nicolas Roche
0003-tests-copy-paste-users-tests-to-member-users-api-508.patch	7,52 ko	23 février 2021	Nicolas Roche
0001-misc-correct-indentation-50889.patch	1,56 ko	23 février 2021	Nicolas Roche
0002-api-add-a-list-role-members-api-50889.patch	11,3 ko	28 février 2021	Nicolas Roche
0003-tests-copy-paste-users-tests-to-member-users-api-508.patch	7,52 ko	28 février 2021	Nicolas Roche
0001-misc-correct-indentation-50889.patch	1,51 ko	28 février 2021	Nicolas Roche
0003-api-add-list-and-retrieve-role-member-s-api-50889.patch	11,7 ko	22 mars 2021	Nicolas Roche
0004-tests-copy-paste-users-tests-to-member-users-api-508.patch	7,51 ko	22 mars 2021	Nicolas Roche
0001-misc-correct-indentation-50889.patch	1,51 ko	22 mars 2021	Nicolas Roche
0002-api-upgrade-change_role-permission-to-manage_members.patch	10,9 ko	22 mars 2021	Nicolas Roche
0002-api-upgrade-change_role-permission-to-manage_members.patch	9,52 ko	30 mars 2021	Nicolas Roche