

Hobo - Bug #69296

sur un appel à /api/check-password/ TypeError: cannot unpack non-iterable NoneType object

19 septembre 2022 20:26 - Sentry lo

Statut:	Fermé	Début:	19 septembre 2022
Priorité:	Normal	Echéance:	
Assigné à:	Emmanuel Cazenave	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

<https://sentry.entrouvert.org/entrouvert/publik/issues/94127/> mais la valeur de authenticator y est cachée,

```
TypeError: cannot unpack non-iterable NoneType object
(8 additional frame(s) were not displayed)
...
File "rest_framework/views.py", line 469, in handle_exception
    self.raise_uncaught_exception(exc)
File "rest_framework/views.py", line 480, in raise_uncaught_exception
    raise exc
File "rest_framework/views.py", line 506, in dispatch
    response = handler(request, *args, **kwargs)
File "authentic2/api_views.py", line 189, in post
    response, response_status = self.rpc(request, serializer)
File "authentic2/api_views.py", line 1399, in rpc
    user, dummy_oidc_client = authenticator.authenticate_credentials(
```

Dans la trace par email on voit <hobo.rest_authentication.APIClientAuthentication object at 0x7fd4a0df18b0>.

Dans le code on a en effet des moments avec "return None" alors que deux valeurs sont attendues.

Demandes liées:

Lié à Authentic 2 - Development #69335: api : respecter l'API DRF dans /api/c... **Solution validé 20 septembre 2022**

Révisions associées

Révision 22c8c2cb - 20 septembre 2022 12:27 - Emmanuel Cazenave

debian: do not load APIClientAuthentication in authentic (#69296)

Historique

#1 - 19 septembre 2022 20:26 - Frédéric Péters

- *Projet changé de Suivi des traces à Hobo*

#2 - 20 septembre 2022 10:23 - Emmanuel Cazenave

- *Assigné à mis à Emmanuel Cazenave*

#3 - 20 septembre 2022 12:29 - Emmanuel Cazenave

- *Fichier 0001-debian-do-not-load-APIClientAuthentication-in-authen.patch ajouté*

- *Statut changé de Nouveau à Solution proposée*

- *Patch proposed changé de Non à Oui*

Boulette dans [#67085](#) : hobo.rest_authentication.APIClientAuthentication ne doit pas être chargé pour authentic (qui se débrouille tout seul via [#66985](#), il ne va pas s'interroger lui même alors qu'il a les objets APIClient dans sa base de donnée).

#4 - 20 septembre 2022 12:35 - Frédéric Péters

Reste que authenticate_credentials est supposé retourner un tuple; dans DRF :

```
def _authenticate(self):
    """
```

```
Attempt to authenticate the request using each authentication instance
in turn.
"""
for authenticator in self.authenticators:
    try:
        user_auth_tuple = authenticator.authenticate(self)
```

(le `.authenticate()` amenant derrière un appel/retour à `authenticate_credentials`)

#5 - 20 septembre 2022 13:03 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#6 - 20 septembre 2022 13:05 - Emmanuel Cazenave

- Statut changé de En cours à Solution proposée

Frédéric Péters a écrit :

Reste que `authenticate_credentials` est supposé retourner un tuple; dans DRF :

J'avais regardé ça et j'en avais pas fait la même lecture (et toujours pas aujourd'hui):

```
def _authenticate(self):
    """
    Attempt to authenticate the request using each authentication instance
    in turn.
    """
    for authenticator in self.authenticators:
        try:
            user_auth_tuple = authenticator.authenticate(self)
        except exceptions.APIException:
            self._not_authenticated()
            raise

    if user_auth_tuple is not None:
        self._authenticator = authenticator
        self.user, self.auth = user_auth_tuple
        return

    self._not_authenticated()
```

Que `authenticate` renvoie `None` c'est le pattern pour permettre de passer à l'authenticator suivant.

Alors ok je peux lever une exception dans `authenticate_credentials` au lieu de renvoyer `None`, et ainsi interrompre l'itération sur les authenticators mais je vois pas l'intérêt. Je m'étais inspiré de `PublikAuthentication.authenticate` qui renvoie `None` s'il ne voit pas de signature dans la requête.

#7 - 20 septembre 2022 13:22 - Benjamin Dauvergne

D'ac alors c'est le code d'authentic qu'il faut corriger, même si ta correction aux settings est bonne, ça ne suit pas l'API DRF.

#8 - 20 septembre 2022 13:57 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

Curieux d'avoir aucune utilisation de cette possibilité de "sauter" un authenticator dans `django-rest-framework` mais ok en effet, et ça ne doit pas poser trop de soucis de manière globale vu que déjà pour l'authentification par signature on faisait ça,

```
def authenticate(self, request):
    full_path = request.get_full_path()
    if not request.GET.get('orig') or not request.GET.get('signature'):
        return None
```

#9 - 20 septembre 2022 14:56 - Emmanuel Cazenave

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

```
commit 22c8c2cb896f71e4b12655d5580714763cc9ae82
Author: Emmanuel Cazenave <ecazenave@entrouvert.com>
Date: Tue Sep 20 12:27:44 2022 +0200
```

```
debian: do not load APIClientAuthentication in authentic (#69296)
```

#10 - 20 septembre 2022 15:02 - Emmanuel Cazenave

- Lié à Development #69335: api : respecter l'API DRF dans /api/check-password/ ajouté

#11 - 20 septembre 2022 23:52 - Transition automatique

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

#12 - 20 novembre 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0001-debian-do-not-load-APIClientAuthentication-in-authen.patch	1,59 ko 20 septembre 2022	Emmanuel Cazenave
---	---------------------------	-------------------