

Mandaye - Development #9702

MandayeJS : construire les cookies a partir de page.cookies et non page.headers + cookie filter

20 janvier 2016 01:10 - Josué Kouka

Statut:	Rejeté	Début:	20 janvier 2016
Priorité:	Normal	Echéance:	
Assigné à:	Josué Kouka	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description			

Historique

#1 - 20 janvier 2016 01:14 - Josué Kouka

- Fichier 0001-build-cookies-from-page.cookies-9702.patch ajouté

#2 - 20 janvier 2016 01:38 - Josué Kouka

- Fichier 0001-build-cookies-from-page.cookies-9702.patch ajouté

- Patch proposed changé de Non à Oui

#3 - 20 janvier 2016 09:32 - Josué Kouka

- Fichier 0001-build-cookies-from-page.cookies-9702.patch ajouté

#4 - 20 janvier 2016 09:58 - Benjamin Dauvergne

Bon le parsing des cookies c'est un peu touchy, j'espère que ta technique va marcher, je serai plus à l'aise en partant d'une base de code sûre (genre requests, webob ou werkzeug); la lib de base Cookie a le problème d'être trop stricte.

Il y a de la lumière au bout du tunnel: <https://cookies.readthedocs.org/en/latest/>

#5 - 20 janvier 2016 10:32 - Frédéric Péters

De mon côté (sur base des expériences passées) je pense qu'il faut au maximum ne pas jouer avec les cookies, qu'il faut transférer les entêtes tels qu'envoyés par l'application, avec le moins de manipulation possible.

À quel besoin répond le changement ?

#6 - 20 janvier 2016 11:59 - Josué Kouka

Frédéric Péters a écrit :

De mon côté (sur base des expériences passées) je pense qu'il faut au maximum ne pas jouer avec les cookies, qu'il faut transférer les entêtes tels qu'envoyés par l'application, avec le moins de manipulation possible.

À quel besoin répond le changement ?

Enfin, je fais une manipulation sans modification. Au départ, je dans phantomjs, je récupérais tous les cookies possible (de peur d'en manquer un qui soit important), puis les chargeait avec SimpleCookie, puis attachait le SimpleCookie à une réponse django.

J'en suis arrivé à le faire parce que quand les cookies récupérer de phantomjs étaient mal formatés (double quotes autour des valeurs, ...), un `response.set_cookie(...)`

setait un cookie avec ces doubles quotes. La solution que j'avais trouvé c'était de les charger sous forme de `raw_cookie` (`SET-COOKIE ...`). C'est ce qui marche sur la recette Vincennes par exemple.

Après 3 implémentations, je me rends compte que peut me fier aux cookies de phantomjs venant de `page.cookies` (page sur laquelle on se trouve quand l'authentification phantomjs est OK). Par contre ces cookies sont sous forme de dictionnaire et il n'y a pas de méthode `SimpleCookie.from_dict()`.

#7 - 20 janvier 2016 12:11 - Josué Kouka

- Fichier `cookies.json` ajouté

Benjamin Dauvergne a écrit :

Bon le parsing des cookies c'est un peu touchy, j'espère que ta technique va marcher, je serai plus à l'aise en partant d'une base de code sûre (genre requests, webob ou werkzeug); la lib de base Cookie a le problème d'être trop stricte.

Il y a de la lumière au bout du tunnel: <https://cookies.readthedocs.org/en/latest/>

Le format de cookies reçu de phantomjs est attachée.

cookies a une méthode une méthode intéressante, mais il me faudrait comme passer par un SimpleCookie afin de pouvoir les attaché à une réponse django.

```
@
from cookies import Cookie
from Cookie import SimpleCookie

def cookie_parser(cookies):
    ck = SimpleCookie()
    for cookie in cookies:
        ck.load(
            Cookie.from_dict(cookie,ignore_bad_attributes=True).render_response()
        )
    return ck
@
render_response qui transforme en "raw_cookie".
```

Le petit souci, c'est qu'il n'y pas de package debian de python-cookies (sauf sur la SID)

#8 - 20 janvier 2016 12:29 - Frédéric Péters

J'en suis arrivé à le faire parce que quand les cookies récupérer de phantomjs étaient mal formatés (double quotes autour des valeurs, ...), un `response.set_cookie(...)` mettait un cookie avec ces doubles quotes. La solution que j'avais trouvé c'était de les charger sous forme de **raw_cookie** (SET-COOKIE ...). C'est ce qui marche sur la recette Vincennes par exemple.

Pour moi il y a des entêtes et il faut les ressortir au client tels qu'ils ont été donnés à phantomjs, sans interprétation ou modification aucune; surtout pas essayer de les remettre carrés, conformes aux specs, ou quoique ce soit. (on a déjà trop galéré par le passé à essayer de comprendre pourquoi une application ne reconnaissait pas ses cookies).

#9 - 20 janvier 2016 13:28 - Benjamin Dauvergne

Je suis d'accord que si phantomjs permet d'avoir les entêtes cookie nus, il faut juste les recopier. Le parsing ne doit avoir lieu que si il y a une manipulation à faire sur les cookies, ça ne semble pas être le cas.

#10 - 20 janvier 2016 13:41 - Josué Kouka

- Statut changé de *En cours* à *Rejeté*

ok ça marche ;)

Fichiers

0001-build-cookies-from-page.cookies-9702.patch	2,92 ko	20 janvier 2016	Josué Kouka
0001-build-cookies-from-page.cookies-9702.patch	3,02 ko	20 janvier 2016	Josué Kouka
0001-build-cookies-from-page.cookies-9702.patch	3,65 ko	20 janvier 2016	Josué Kouka
cookies.json	4,73 ko	20 janvier 2016	Josué Kouka