

w.c.s. - Bug #20523

crash 500 lors de l'affichage d'une source de donnée avec une mauvaise URL

08 décembre 2017 09:28 - Thomas Noël

Statut:	Fermé	Début:	08 décembre 2017
Priorité:	Haut	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		

Description

Une source de données avec une URL "/csvdatasource/services/data" donne cette trace 500 quand elle est affichée :

Exception:

```
type = '<type 'exceptions.TypeError'>', value = 'argument of type 'NoneType' is not iterable'
```

Stack trace (most recent call first):

```
File "/usr/lib/python2.7/dist-packages/wcs/qommon/misc.py", line 284, in _http_request
 282
 283     auth = None
> 284     if '@' in hostname:
 285         authenticator, hostname = hostname.split('@')
 286         if ':' in authenticator:
```

locals:

```
body = None
url = '/csvdatasource/services/data'
cert_file = None
hostname = None
auth = None
raise_on_http_errors = True
headers = {}
timeout = None
query = 'tasource/services/data'
method = 'GET'
```

```
File "/usr/lib/python2.7/dist-packages/wcs/qommon/misc.py", line 314, in urlopen
 312     url, 'GET' if data is None else 'POST',
 313     body=data,
> 314     raise_on_http_errors=True)
 315     return StringIO(data)
 316
```

locals:

```
url = '/csvdatasource/services/data'
data = None
```

```
File "/usr/lib/python2.7/dist-packages/wcs/data_sources.py", line 183, in get_structured_items
 181     get_logger().warn('Error loading JSON data source (%s)' % str(e))
 182     except ValueError as e:
> 183     get_logger().warn('Error reading JSON data source output (%s)' % str(e))
 184     return []
 185
```

locals:

```
url = '/csvdatasource/services/data'
data_source = {'type': 'json', 'value': '/csvdatasource/services/data'}
```

```
File "/usr/lib/python2.7/dist-packages/wcs/admin/data_sources.py", line 128, in _q_index
 126
 127     if data_source_type in ('json', 'formula'):
> 128     items = get_structured_items(self.datasources.data_source)
```

```
129         if items:
130             r += htmltext('<h3>%s</h3>') % _('Preview (first items only)')
...

```

Révisions associées

Révision ad6849f5 - 15 décembre 2017 16:38 - Thomas Noël

misc: accept only http and https as URL scheme (#20523)

Historique

#2 - 08 décembre 2017 10:25 - Thomas Noël

Il y a mystère pour moi ici, car rien ne plante si j'ajoute ceci à tests/test_admin_pages.py

```
3890 def test_data_sources_view(pub):
...
3947     data_source.data_source = {'type': 'json', 'value': '/csvdatasource/services/data'}
3948     data_source.store()
3949     resp = app.get('/backoffice/settings/data-sources/%s/' % data_source.id)

```

#3 - 08 décembre 2017 10:52 - Benjamin Dauvergne

N'y aurait pas remplacement de `_http_request` dans les tests ? pour mocker les réponses.

Dans tous les cas faut réécrire ça:

```
if url.startswith('http://'):
    hostname, query = urllib.splithost(url[5:])
else:
    hostname, query = urllib.splithost(url[6:])

```

sur la base de `urlparse.urlparse`, c'est juste pas bon comme code.

#4 - 08 décembre 2017 11:51 - Thomas Noël

Benjamin Dauvergne a écrit :

N'y aurait pas remplacement de `_http_request` dans les tests ? pour mocker les réponses.

C'est bien possible mais par quelle magie ? Je ne vois rien dans `test_admin_pages.py` qui parle de mocker quoi que ce soit...

Dans tous les cas faut réécrire ça:

(...)

sur la base de `urlparse.urlparse`, c'est juste pas bon comme code.

Oui, après avoir corrigé ça, c'est en cherchant à écrire un test que je bloque.

#5 - 08 décembre 2017 12:12 - Benjamin Dauvergne

Thomas Noël a écrit :

Benjamin Dauvergne a écrit :

N'y aurait pas remplacement de `_http_request` dans les tests ? pour mocker les réponses.

C'est bien possible mais par quelle magie ? Je ne vois rien dans `test_admin_pages.py` qui parle de mocker quoi que ce soit...

C'est global ça ne passe pas par une fixture (!?!, [#16509](#) c'est corrigé depuis longtemps, je suis un peu malheureux avec mes tickets).

Dans tous les cas faut réécrire ça:

(...)

sur la base de `urlparse.urlparse`, c'est juste pas bon comme code.

Oui, après avoir corrigé ça, c'est en cherchant à écrire un test que je bloque.

Ben intègre le [#16509](#) et tu pourras.

#6 - 08 décembre 2017 16:20 - Thomas Noël

- Fichier 0001-misc-accept-only-http-and-https-as-URL-scheme-20523.patch ajouté
- Statut changé de Nouveau à En cours
- Patch proposed changé de Non à Oui

Après [#16509](#) on peut donc avoir le patch ci-joint

#7 - 15 décembre 2017 10:55 - Thomas Noël

- Priorité changé de Normal à Haut

Bogue grave : le problème remonte y compris sur l'affichage backoffice d'un formulaire où un champ fait référence à une telle source de données. Ce qui peut facilement arriver, typiquement une source de donnée avec une URL [form_var_chose] où la variable n'existe pas encore...

#10 - 15 décembre 2017 16:18 - Benjamin Dauvergne

Ack, idéalement il faudrait valider qu'on obtient un début d'URL dans wcs.data_sources.DataSourceSelectionWidget quand on choisit JSON ou JSONP, je te laisse ouvrir un ticket si ça te semble pertinent.

#11 - 15 décembre 2017 16:36 - Thomas Noël

Benjamin Dauvergne a écrit :

Ack, idéalement il faudrait valider qu'on obtient un début d'URL dans wcs.data_sources.DataSourceSelectionWidget quand on choisit JSON ou JSONP, je te laisse ouvrir un ticket si ça te semble pertinent.

Pas vraiment parce que l'URL peut être une [form_var...] qui dépend d'une variable d'un formulaire, et ne sera résolue que dans ce contexte. C'est même, la plupart du temps, ce qui donne l'erreur 500 qui est ici contournée.

#12 - 15 décembre 2017 16:40 - Thomas Noël

- Statut changé de En cours à Résolu (à déployer)

```
commit ad6849f523ff512dc8c7226fad45fd132ddbc31f (HEAD -> master)
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Fri Dec 8 16:19:04 2017 +0100
```

```
misc: accept only http and https as URL scheme (#20523)
```

#13 - 15 décembre 2017 19:02 - Benjamin Dauvergne

Thomas Noël a écrit :

Benjamin Dauvergne a écrit :

Ack, idéalement il faudrait valider qu'on obtient un début d'URL dans wcs.data_sources.DataSourceSelectionWidget quand on choisit JSON ou JSONP, je te laisse ouvrir un ticket si ça te semble pertinent.

Pas vraiment parce que l'URL peut être une [form_var...] qui dépend d'une variable d'un formulaire, et ne sera résolue que dans ce contexte. C'est même, la plupart du temps, ce qui donne l'erreur 500 qui est ici contournée.

En général le début de l'URL c'est plutôt [passerelle_url] qui est toujours disponible, a-t-on un seul cas où la partie host de l'URL dépend du formulaire ?

#14 - 15 décembre 2017 23:36 - Thomas Noël

Benjamin Dauvergne a écrit :

En général le début de l'URL c'est plutôt [passerelle_url] qui est toujours disponible, a-t-on un seul cas où la partie host de l'URL dépend du formulaire ?

Oui, pour les réservations et rendez-vous, après le choix d'un type de rdv on affiche la liste des dispos avec [form_var_type_rdv_url] (c'est même une des raisons de l'envoi de ce patch en hotfix)

#15 - 23 décembre 2018 14:44 - Frédéric Péters

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

