

Passerelle - Bug #39391

cmis py3: pas bon les bytes dans StringIO

29 janvier 2020 16:04 - Emmanuel Cazenave

Statut:	Fermé	Début:	29 janvier 2020
Priorité:	Normal	Echéance:	
Assigné à:	Nicolas Roche	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Je n'ai plus la trace mais en py3 on essaie de passer des bytes dans un StringIO, crash ici :			
<pre>@wrap_cmis_error def create_doc(self, file_name, file_path, file_byte_content): folder = self._get_or_create_folder(file_path) return folder.createDocument(file_name, contentFile=StringIO(file_byte_content))</pre>			
Les tests unitaires court-circuitent ce chemin de code.			
Demandes liées:			
Lié à Passerelle - Development #39597: cmis : utiliser httmock dans les tests...		Fermé	06 février 2020

Révisions associées

Révision 1480f038 - 06 février 2020 15:19 - Nicolas Roche

cmis: allow to upload binary documents (#39391)

Historique

#1 - 03 février 2020 18:30 - Nicolas Roche

- Assigné à mis à Nicolas Roche

#2 - 04 février 2020 07:42 - Emmanuel Cazenave

Emmanuel Cazenave a écrit :

Les tests unitaires court-circuitent ce chemin de code.

Et donc ça pourrait être l'occasion de revoir les tests unitaires pour qu'ils ne fassent plus ce court-circuit (je dis ça en passant mais si c'est trop galère, ça peut-être l'objet d'un ticket à part).

#3 - 04 février 2020 14:56 - Nicolas Roche

- Fichier 0001-cmis-allow-to-upload-binary-documents-39391.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Comment savoir si ça ne va pas crasher plus loin ?

#4 - 04 février 2020 15:02 - Benjamin Dauvergne

Nicolas Roche a écrit :

Comment savoir si ça ne va pas crasher plus loin ?

Suffit de mocker de la réponse HTTP plutôt que le retour de createDocument.

#5 - 04 février 2020 20:02 - Nicolas Roche

Mais ensuite, ça pourrai encore crasher à l'envoi.

(je peux tester en vrai quelque-part, pour savoir quoi mettre dans le mock ?)

#6 - 04 février 2020 20:58 - Benjamin Dauvergne

Nicolas Roche a écrit :

Mais ensuite, ça pourrai encore crasher à l'envoi.
(je peux tester en vrai quelque-part, pour savoir quoi mettre dans le mock ?)

Je ne vois pas comment, si ça passe requests, c'est fini, ça marche. Utilise httmock, il y a des exemples ailleurs dans le code de passerelle.

```
tests/confstest.py:2:from httmock import urlmatch, HTTPMock, response
tests/test_api_entreprise.py:23:from httmock import urlmatch, HTTPMock, response
tests/test_api_particulier.py:20:from httmock import urlmatch, HTTPMock, response
tests/test_availability.py:4:from httmock import all_requests, HTTPMock
tests/test_cartads_cs.py:9:from httmock import HTTPMock
tests/test_dpark.py:9:import httmock
tests/test_lille_urban_card.py:3:from httmock import HTTPMock
tests/test_mdph13.py:26:import httmock
tests/test_planitech.py:6:from httmock import urlmatch, HTTPMock
tests/test_proxylogger.py:8:from httmock import HTTPMock
tests/test_requests.py:6:from httmock import urlmatch, HTTPMock, response
tests/test_strasbourg_eu.py:3:from httmock import all_requests, HTTPMock
tests/utills.py:4:import httmock
tests/wcs/confstest.py:32:import httmock
```

#7 - 05 février 2020 09:34 - Valentin Deniaud

Nicolas Roche a écrit :

(je peux tester en vrai quelque-part, pour savoir quoi mettre dans le mock ?)

Vouli https://dev.entrouvert.org/projects/sysadmin/wiki/CMIS_-_Apache_Chemistry (merci Manu)

#8 - 06 février 2020 14:35 - Valentin Deniaud

Cependant (et parce que mes tests sont rouges dans [#39416](#) à cause de ce bug),

Emmanuel Cazenave a écrit :

Les tests unitaires court-circuitent ce chemin de code.

Pas vraiment, comme le montre le test de Nico, le chemin est couvert mais mal, et ça n'a pas à voir avec un mock de la réponse du serveur CMIS. Donc l'amélioration des tests en mockant les vraies réponses devrait être faite dans un autre ticket (ou pas du tout).

Nico, je ack si tu me remplaces la condition sur les bytes par un bon vieux `force_str` :

#9 - 06 février 2020 15:03 - Nicolas Roche

- Lié à [Development #39597](#): *cmis : utiliser httmock dans les tests pour avoir une couverture 'plus profonde' ajouté*

#10 - 06 février 2020 15:08 - Nicolas Roche

- *Fichier 0001-cmis-allow-to-upload-binary-documents-39391.patch ajouté*

Merci pour les info pour tester "en vrai".

Je relaye la proposition de Benjamin de compléter les tests dans [#39597](#), afin de ne pas bloquer Valentin.

Patch qui prend simplement en compte le qui qu'on s'attend ici à recevoir du binaire :

```
data['file_byte_content'] = base64.b64decode(file_['content'])
```

or

```
$ pydoc base64
decode(input, output)
Decode a file; input and output are binary files.
```

#11 - 06 février 2020 15:17 - Valentin Deniaud

- *Statut changé de Solution proposée à Solution validée*

Parfait.

#12 - 06 février 2020 15:20 - Nicolas Roche

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

```
commit 1480f03881764a6a3a488d5c3a823c5f109f167a
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Thu Feb 6 14:54:32 2020 +0100
```

```
cmis: allow to upload binary documents (#39391)
```

#13 - 07 février 2020 09:34 - Frédéric Péters

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

Fichiers

0001-cmis-allow-to-upload-binary-documents-39391.patch	2,64 ko	04 février 2020	Nicolas Roche
0001-cmis-allow-to-upload-binary-documents-39391.patch	2,11 ko	06 février 2020	Nicolas Roche