Passerelle - Development #38130

UnicodeEncodeError at /dpark/parking/link/

02 décembre 2019 14:28 - Nicolas Roche

Statut:	Fermé	**Début:**	02 décembre 2019
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

(à priori) Problème pour afficher le message d'erreur retourné par DPark.

```
Exception Type:     UnicodeEncodeError
Exception Value:     'ascii' codec can't encode character u'\xe9' in position 47: ordinal not in r
ange(128)
Exception Location:     /usr/lib/python2.7/dist-packages/passerelle/contrib/dpark/models.py in cal
l, line 205

args :
(<WSGIRequest: POST '/dpark/parking/link/?orig=demarches-montoulouse.eservices.toulouse-metropole.
fr&algo=sha256&timestamp=2019-12-02T13%3A12%3A58Z&nonce=5fd51e97dd1566faf252dc13427f8278&signature
=QWIxYrfc3anm/17kfE2ur%2BCkqJIdlEzqawQIEVpiPyE%3D'>,)

e :
UnicodeEncodeError('ascii', u"La valeur 'E-6-VMQN504M' ne respecte pas le sch\xe9ma XSD.", 47, 48,
 'ordinal not in range(128)')

extras :
{'body': '\'{"lastname": "LANNELONGUE", "cardnumber": "", "firstnames": "VINCENT", "badgenumber":
"", "filenumber": "E-6-VMQN504M", "nameid": "8da5dc154c544647ba33cc9d84db88b7"}\'',
 'exception': u"'ascii' codec can't encode character u'\\xe9' in position 47: ordinal not in range
(128)",
```

Avec un peu de chance c'est déjà résolu par #37880

Révisions associées

Révision 821cc213 - 16 janvier 2020 15:45 - Nicolas Roche

dpark: use only unicode strings (#38130)

Historique

#1 - 02 décembre 2019 16:30 - Thomas Noël

L'erreur se produit ligne 205 :

```
198     def call(self, operation, *args, **kwargs):
199         client = get_client(self)
200         proxy_service = client.create_service(SERVICE_BINDING, self.operation_url)
201         bypass_erroneous_reply = kwargs.pop('bypass_erroneous_reply', False)
202         try:
203             reply = getattr(proxy_service, operation)(*args, **kwargs)
204         except (WebFault, ) as exc:
205             raise APIError('ServiceError: %s' % exc.message)
206         except (Exception,) as exc:
207             raise APIError('Error: %s' % exc.message)
208         reply_code = getattr(reply, 'CodeRetour', None) or getattr(reply, 'Code_Retour', None)
209         reply_message = getattr(reply, 'MessageRetour', None) or getattr(reply, 'Lib_Retour', None)
210         if reply_code != '01' and not bypass_erroneous_reply:
211             raise APIError(reply_message)
212         return reply
```

C'est DPark qui n'est pas content et renvoie le message avec un encodage latin1. Il faudrait peut-être passer à %r sur les exc.message.

#2 - 16 décembre 2019 10:35 - Benjamin Dauvergne

force_text(exc.message , 'latin1', error='ignore')

#3 - 19 décembre 2019 16:19 - Nicolas Roche

- Fichier 0001-dpark-accept-error-message-containing-latin1-38130.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

J'avoue que j'ai du mal à comprendre :

- si dans le test je met "éêè" à la place de u"éêè", je n'ai plus l'erreur
- et si je met u"éêè".encode('latin_1'), alors j'ai DjangoUnicodeDecodeError au décodage

Néanmoins, je pense avoir réussi à reproduire l'erreur de la trace.

#4 - 15 janvier 2020 14:06 - Thomas Noël

(je suis perdu dans la compréhension du code et du test, l'encodage et moi... je passe la main)

#5 - 15 janvier 2020 19:52 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

Thomas Noël a écrit :

> (je suis perdu dans la compréhension du code et du test, l'encodage et moi... je passe la main)

En fait c'est ton analyse initiale qui est mauvaise, il n'y a pas de latin1 ici, zeep décode bien le message et en fait de l'unicode, c'est juste qu'on fait un

```
'%s' % exc
```

et Exception ne gère pas ça :

```
>>> str(Exception(u'é'))
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
UnicodeEncodeError: 'ascii' codec can't encode character u'\xe9' in position 0: ordinal not in range(128)
```

Un simple

```
% exc.message
```

suffira.

#6 - 16 janvier 2020 00:08 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#7 - 16 janvier 2020 00:08 - Benjamin Dauvergne

- Fichier 0001-dpark-use-only-unicode-strings-38130.patch ajouté

- Tracker changé de Bug à Development

- Statut changé de En cours à Solution proposée

Voilà on passe tout en unicode et ça marche mieux (en entrée on a que de l'unicode et en sortie aussi via zeep), et afficher une exception avec de l'unicode dans une chaîne unicode ça marche par contre.

#8 - 16 janvier 2020 09:51 - Nicolas Roche

Je ne connaissais pas https://python-future.org/unicode_literals.html
ça fait le job, et ça corrige mon oubli sur la seconde exception.
Pour moi c'est bon, mais je laisse la main.

#9 - 16 janvier 2020 12:22 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

#10 - 16 janvier 2020 15:46 - Benjamin Dauvergne

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

```
commit 19694e1a47ee1082876cc37b5e400bd05309a3f5
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date:   Thu Dec 19 15:57:27 2019 +0100

    dpark: use only unicode strings (#38130)
```

#11 - 19 janvier 2020 16:15 - Frédéric Péters

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

Fichiers

0001-dpark-accept-error-message-containing-latin1-38130.patch	2,58 ko	19 décembre 2019	Nicolas Roche
0001-dpark-use-only-unicode-strings-38130.patch	2,03 ko	15 janvier 2020	Benjamin Dauvergne