

## Combo - Bug #14627

### trace sur la récup de facture lors du callback du paiement

17 janvier 2017 07:18 - Frédéric Péters

<b>Statut:</b>	Fermé	<b>Début:</b>	17 janvier 2017
<b>Priorité:</b>	Haut	<b>Echéance:</b>	
<b>Assigné à:</b>		<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	
<b>Patch proposed:</b>	Oui		

**Description**

<https://sentry.entrouvert.org/sentry/production/issues/1194/>

```
Internal Server Error: /lingo/callback/1/
Traceback (most recent call last):
  File "/usr/lib/python2.7/dist-packages/django/core/handlers/base.py", line 111, in get_response
    response = wrapped_callback(request, *callback_args, **callback_kwargs)
  File "/usr/lib/python2.7/dist-packages/django/views/generic/base.py", line 69, in view
    return self.dispatch(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/django/views/decorators/csrf.py", line 57, in wrapped_view
    return view_func(*args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/combo/apps/lingo/views.py", line 452, in dispatch
    return super(CallbackView, self).dispatch(*args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/django/views/generic/base.py", line 87, in dispatch
    return handler(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/combo/apps/lingo/views.py", line 445, in get
    return self.handle_callback(request, request.environ['QUERY_STRING'], **kwargs)
  File "/usr/lib/python2.7/dist-packages/combo/apps/lingo/views.py", line 432, in handle_callback
    remote_item = regie.get_invoice(user=transaction.user, invoice_id=item_id)
  File "/usr/lib/python2.7/dist-packages/combo/apps/lingo/models.py", line 143, in get_invoice
    response.raise_for_status()
  File "/usr/lib/python2.7/dist-packages/requests/models.py", line 825, in raise_for_status
    raise HTTPError(http_error_msg, response=self)
HTTPError: 500 Server Error: INTERNAL SERVER ERROR
```

Il y a sans doute une facture à manuellement marquer comme payée. Et réfléchir à comment gérer ça.

#### Révisions associées

##### Révision 0665cb8b - 18 juin 2017 11:00 - Benjamin Dauvergne

lingo: notify remote\_item of payments asynchronously (#14627)

In case the remote\_item cannot be notified synchronously of a payment, we keep around a list of items to notify in the Transaction.to\_be\_paid\_remote\_items field. They will be notified by the update\_transactions command launched by cron.

#### Historique

##### #1 - 17 janvier 2017 20:18 - Benjamin Dauvergne

À coté de cela je ne trouve pas la trace coté passerelle, ni dans syslog ni dans sentry, ce qui m'étonne.

##### #2 - 17 janvier 2017 20:19 - Benjamin Dauvergne

My bad les heures sur sentry sont UTC.

```
Jan 17 00:31:33 passerelle passerelle ERROR passerelle.alfortville.fr 5.135.221.12 - r:7F5BEB3D7250 Error occurred while processing request
```

Traceback (most recent call last):

```
File "/usr/lib/python2.7/dist-packages/passerelle/utils/jsonresponse.py", line 372, in api
    resp = f(*args, **kwargs)
File "/usr/lib/python2.7/dist-packages/passerelle/utils/__init__.py", line 135, in _wrapped_view
```

```

return view_func(instance, request, *args, **kwargs)
File "/usr/lib/python2.7/dist-packages/passerelle/contrib/agoraplus/views.py", line 124, in get
return self.get_data(request, *args, **kwargs)
File "/usr/lib/python2.7/dist-packages/passerelle/contrib/agoraplus/views.py", line 541, in get_data
return self.object.get_invoice(self.login, invoice_id)
File "/usr/lib/python2.7/dist-packages/passerelle/contrib/agoraplus/models.py", line 835, in get_invoice
for i in self.get_invoices(login):
File "/usr/lib/python2.7/dist-packages/passerelle/contrib/agoraplus/models.py", line 813, in get_invoices
with wrap_agora_dict(self.get_user(login), 'to get user') as user:
File "/usr/lib/python2.7/dist-packages/passerelle/contrib/agoraplus/models.py", line 200, in get_user
return self.request(endpoint)
File "/usr/lib/python2.7/dist-packages/passerelle/contrib/agoraplus/models.py", line 174, in request
response = self.request_auth(endpoint, **kwargs)
File "/usr/lib/python2.7/dist-packages/passerelle/contrib/agoraplus/models.py", line 145, in request_auth
response = self.requests.request(method, self.url + endpoint, **kwargs)
File "/usr/lib/python2.7/dist-packages/passerelle/utils/__init__.py", line 155, in request
response = super(LoggedRequest, self).request(method, url, **kwargs)
File "/usr/lib/python2.7/dist-packages/requests/sessions.py", line 457, in request
resp = self.send(prepare, **send_kwargs)
File "/usr/lib/python2.7/dist-packages/requests/sessions.py", line 569, in send
r = adapter.send(request, **kwargs)
File "/usr/lib/python2.7/dist-packages/requests/adapters.py", line 407, in send
raise ConnectionError(err, request=request)
ConnectionError: ('Connection aborted.', gaierror(-2, 'Name or service not known'))

```

OVH a encore frappé.

### #3 - 17 janvier 2017 21:08 - Benjamin Dauvergne

- Fichier 0001-notify-remote\_item-of-payments-asynchronously-fixes-.patch ajouté
- Patch proposed changé de Non à Oui

Pour la correction en base de la facture d'Alfortville je laisse les gens qui connaissent dire ce qu'il faut faire.

### #4 - 29 mai 2017 17:39 - Frédéric Péters

- Fichier 0001-lingo-notify-remote\_item-of-payments-asynchronously-.patch ajouté

Mise à jour du patch (app.get qui exige désormais un paramètre nommé).

Aussi, plutôt que :

```

+         if self.to_be_paid_remote_items or to_be_paid_remote_items:
+             self.to_be_paid_remote_items = ','.join(to_be_paid_remote_items) or None
+             self.save(update_fields=['to_be_paid_remote_items'])

```

simplement :

```

+         self.to_be_paid_remote_items = ','.join(to_be_paid_remote_items) or None
+         self.save(update_fields=['to_be_paid_remote_items'])

```

que je trouve plus rassurant à comprendre.

### #5 - 30 mai 2017 08:09 - Frédéric Péters

- Fichier 0001-lingo-notify-remote\_item-of-payments-asynchronously-.patch ajouté
- Statut changé de Nouveau à En cours

Et la migration réordonnée. (tapée après celle de [#16065](#)).

### #6 - 18 juin 2017 03:08 - Thomas Noël

Dans un mode parano, on remplirait d'abord `to_be_paid_remote_items` avec tous les items, on bouclerait dessus, et on réduirait le `self.to_be_paid_remove_items` à chaque paiement réussi (avec un `self.save(update_fields=['to_be_paid_remote_items'])` à chaque fois)

Bon, après, je ne vois pas trop ce qui pourrait planter dans le "else:".

Donc, ack, sauf si un des auteurs dit que j'ai trop raison.

### #7 - 18 juin 2017 11:02 - Frédéric Péters

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

Poussé ainsi.

```
commit 0665cb8bbdb5aa149cf84f1ec99aa92adbb1b7
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Tue Jan 17 21:05:23 2017 +0100
```

```
lingo: notify remote_item of payments asynchronously (#14627)
```

```
In case the remote_item cannot be notified synchronously of a payment, we keep
around a list of items to notify in the Transaction.to_be_paid_remote_items
field. They will be notified by the update_transactions command launched by
cron.
```

#### #8 - 23 décembre 2018 15:22 - Frédéric Péters

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

#### Fichiers

---

0001-notify-remote_item-of-payments-asynchronously-fixes-.patch	10,3 ko	17 janvier 2017	Benjamin Dauvergne
0001-lingo-notify-remote_item-of-payments-asynchronously-.patch	10,3 ko	29 mai 2017	Frédéric Péters
0001-lingo-notify-remote_item-of-payments-asynchronously-.patch	9,88 ko	30 mai 2017	Frédéric Péters