

**UnreadablePostError: error during read(65536) on wsgi.input**

04 septembre 2017 10:04 - Frédéric Péters

<b>Statut:</b>	Fermé	<b>Début:</b>	04 septembre 2017
<b>Priorité:</b>	Normal	<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>	Non
<b>Patch proposé:</b>	Non		
<b>Description</b>			
À investiguer, peut-être normal, genre interruption par le client.			
<pre>Internal Server Error: /tmp-upload ... File "/usr/lib/python2.7/dist-packages/django/http/request.py", line 261, in _load_post_and_file s     self._post, self._files = self.parse_file_upload(self.META, data) File "/usr/lib/python2.7/dist-packages/django/http/request.py", line 226, in parse_file_upload     return parser.parse() File "/usr/lib/python2.7/dist-packages/django/http/multipartparser.py", line 209, in parse     for chunk in field_stream: File "/usr/lib/python2.7/dist-packages/django/utils/six.py", line 558, in next     return type(self).__next__(self) File "/usr/lib/python2.7/dist-packages/django/http/multipartparser.py", line 356, in __next__     output = next(self._producer) File "/usr/lib/python2.7/dist-packages/django/utils/six.py", line 558, in next     return type(self).__next__(self) File "/usr/lib/python2.7/dist-packages/django/http/multipartparser.py", line 487, in __next__     for bytes in stream: File "/usr/lib/python2.7/dist-packages/django/utils/six.py", line 558, in next     return type(self).__next__(self) File "/usr/lib/python2.7/dist-packages/django/http/multipartparser.py", line 356, in __next__     output = next(self._producer) File "/usr/lib/python2.7/dist-packages/django/utils/six.py", line 558, in next     return type(self).__next__(self) File "/usr/lib/python2.7/dist-packages/django/http/multipartparser.py", line 418, in __next__     data = self.flo.read(self.chunk_size) File "/usr/lib/python2.7/dist-packages/django/http/request.py", line 295, in read     six.reraise(UnreadablePostError, UnreadablePostError(*e.args), sys.exc_info()[2]) File "/usr/lib/python2.7/dist-packages/django/http/request.py", line 293, in read     return self._stream.read(*args, **kwargs) File "/usr/lib/python2.7/dist-packages/django/core/handlers/wsgi.py", line 57, in read     result = self.buffer + self._read_limited(size - len(self.buffer)) File "/usr/lib/python2.7/dist-packages/django/core/handlers/wsgi.py", line 45, in _read_limited     result = self.stream.read(size) UnreadablePostError: error during read(65536) on wsgi.input</pre>			
<b>Demandes liées:</b>			
Lié à Hobo - Bug #21423: logging: considérer les erreurs UnreadablePostError ...		<b>Nouveau</b>	<b>25 janvier 2018</b>

**Historique****#1 - 04 septembre 2017 10:15 - Thomas Noël**

A priori c'est ça, code "499" signalé par nginx :

```
176.x.x.x - - [04/Sep/2017:10:05:20 +0200] "POST /tmp-upload HTTP/1.1" 499 0 "https://demarches.chateauroux-me
tropole.fr/piscine-a-vagues/inscription-aquagym-piscine-a-vagues-1/" "Mozilla/5.0 (Windows NT 10.0; WOW64; rv:
55.0) Gecko/20100101 Firefox/55.0"
```

**#2 - 10 octobre 2017 14:24 - Thomas Noël**

Juste pour ajouter du contexte, la trace complète est vue dans le journal de wcs :

```
oct. 10 13:39:59 auquo uwsgi[23594]: [uwsgi-body-read] Error reading 40960 bytes. Content-Length: 897426 consumed: 155648 left: 741778 message: Client closed connection
oct. 10 13:39:59 auquo uwsgi[27700]: wsgi ERROR - 89.88.135.182 - r:7F2BD6922390 Internal Server Error: /tmp-upload
Traceback (most recent call last):
  File "/usr/lib/python2.7/dist-packages/django/core/handlers/base.py", line 108, in get_response
    response = middleware_method(request)
  File "/usr/lib/python2.7/dist-packages/wsgi/middleware.py", line 46, in process_request
    pub.parse_request(compat_request)
  File "/usr/lib/python2.7/dist-packages/quixote/publish.py", line 113, in parse_request
    ...
oct. 10 13:40:00 auquo uwsgi[23594]: Tue Oct 10 13:40:00 2017 - SIGPIPE: writing to a closed pipe/socket/fd (probably the client disconnected) on request /tmp-upload (ip 0.0.0.0) !!!
oct. 10 13:40:00 auquo uwsgi[23594]: Tue Oct 10 13:40:00 2017 - uwsgi_response_writev_headers_and_body_do(): Broken pipe [core/writer.c line 287] during POST /tmp-upload (0.0.0.0)
oct. 10 13:40:00 auquo uwsgi[23594]: IOError: write error
oct. 10 13:40:00 auquo uwsgi[23594]: [pid: 27700|app: 0|req: 81140/591362] 0.0.0.0 () {52 vars in 1168 bytes} [Tue Oct 10 13:39:59 2017] POST /tmp-upload => generated 0 bytes in 801 msec (HTTP/1.0 500) 1 headers in 0 bytes (0 switches on core 0)
```

### #3 - 24 janvier 2018 19:22 - Thomas Noël

Un peu d'investigation, c'est en fait totalement "normal" : <https://docs.djangoproject.com/fr/2.0/ref/exceptions/#unreadableposterror>

Reste à savoir si on veut juste :

- ne plus recevoir d'alerte, et donc régler le logging pour ça, comme indiqué ici : <https://docs.djangoproject.com/fr/2.0/topics/logging/#id5>
- gérer ça dans le code de `/usr/lib/python2.7/dist-packages/wsgi/compat.py` (mais gérer quoi comment, je sais pas trop)

### #4 - 24 janvier 2018 20:47 - Benjamin Dauvergne

Je serai pour faire effectivement un filtre au niveau des logs et convertir ça en simple warning.

### #5 - 25 janvier 2018 01:55 - Thomas Noël

Benjamin Dauvergne a écrit :

Je serai pour faire effectivement un filtre au niveau des logs et convertir ça en simple warning.

Je vois très bien comment filtrer (ne rien faire de ces erreurs) mais pour la conversion, dans un logging.Filter ça m'a l'air délicat de changer le `record.level`, c'est peut-être un peu trop tard et ça peut rendre le routage du message incohérent, je pense (la doc semble dire que faut éviter, <https://docs.python.org/2/library/logging.html#filter-objects> " Obviously changing the LogRecord needs to be done with some care, ...")

Tu vois une autre idée ? (filtrer simplement ne me poserait pas de gros gros soucis, on a déjà des warning sur nginx et uwsgi)

### #6 - 25 janvier 2018 09:29 - Benjamin Dauvergne

Thomas Noël a écrit :

Benjamin Dauvergne a écrit :

Je serai pour faire effectivement un filtre au niveau des logs et convertir ça en simple warning.

Je vois très bien comment filtrer (ne rien faire de ces erreurs) mais pour la conversion, dans un logging.Filter ça m'a l'air délicat de changer le `record.level`, c'est peut-être un peu trop tard et ça peut rendre le routage du message incohérent, je pense (la doc semble dire que faut éviter, <https://docs.python.org/2/library/logging.html#filter-objects> " Obviously changing the LogRecord needs to be done with some care, ...")

Je le fais déjà sur authentic: [http://git.entrouvert.org/authentic.git/tree/src/authentic2/log\\_filters.py#n31](http://git.entrouvert.org/authentic.git/tree/src/authentic2/log_filters.py#n31)

Filtrer au niveau du logger c'est sans souci, c'est fait avant la prise en charge du record par le reste de la machinerie dans logging.

Tu vois une autre idée ? (filtrer simplement ne me poserait pas de gros gros soucis, on a déjà des warning sur nginx et uwsgi)

Comme tu veux.

**#7 - 25 janvier 2018 13:35 - Thomas Noël**

- Lié à Bug #21423: logging: considérer les erreurs UnreadablePostError en warning seulement ajouté

**#8 - 21 juillet 2023 13:46 - Frédéric Péters**

- Statut changé de Nouveau à Fermé

- Planning mis à Non

Je ne trouve pas d'occurrence récente de ce type d'erreur.