

w.c.s. - Bug #67834

pas de remontée sentry sur des erreurs de transaction (exception psycopg2.InternalError)

29 juillet 2022 17:41 - Thomas Noël

Statut:	Fermé	Début:	29 juillet 2022
Priorité:	Normal	Echéance:	
Assigné à:	Nicolas Roche	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposé:	Oui		

Description

Vu tout à l'heure, des traces reçues par mail :

Exception:

```
type = '<class 'psycopg2.InternalError'>', value = 'current transaction is aborted, commands ignored until end of transaction block'
```

Stack trace (most recent call first):

```
File "/usr/lib/python3/dist-packages/wcs/sql.py", line 2670, in store
2668         ' AND '.join(where_clauses),
2669     )
> 2670     cur.execute(sql_statement, sql_dict)
2671     if cur.fetchone() is None:
2672         if len(where) > 1:
```

qui n'ont pas donné lieu à des messages dans sentry.

Ce ticket pour comprendre et boucher ce trou.

Révisions associées

Révision b761fbf1 - 30 août 2022 15:27 - Nicolas Roche

misc: send frontoffice exceptions to sentry too (#67834)

Historique

#2 - 05 août 2022 11:58 - Nicolas Roche

- Fichier 0001-compat-add-foreground-transaction-error-to-sentry-67.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Assigné à mis à Nicolas Roche

- Patch proposed changé de Non à Oui

Pour moi les sentry ne sont envoyées que depuis le backoffice.

Voici un patch pour avoir aussi cette exception traitée en front,

testé via cette sentry :

<https://sentry.entrouvert.org/entrouvert/publik/issues/62683>

#3 - 05 août 2022 12:17 - Benjamin Dauvergne

Je ne vois pas pourquoi tu spécialises ainsi, en fait on a aucune exception ne remonte dans sentry dès qu'elle passe dans le cas Exception ici, c'est le cas Exception qu'il faut chopper.

Actuellement ça remonte (faut regarder le champ mechanism) :

- quand c'est dans un thread
- dans le cas template géré à la main
- dans le ou ça arrive après comme un middleware django, tant que ça remonte au handler de requête c'est bon

En fait on en manque un bon paquet (et il y a aussi tout ce qui est logger avec logged_errors qui ne remonte pas, mais ça c'est plus délicat).

#4 - 05 août 2022 12:17 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#5 - 05 août 2022 12:31 - Thomas Noël

Benjamin Dauvergne a écrit :

En fait on en manque un bon paquet (et il y a aussi tout ce qui est logger avec `logged_errors` qui ne remonte pas, mais ça c'est plus délicat).

C'est assez normal à mon sens : les `logged_errors` sont destinées aux administrateurs fonctionnels et ne sont donc pas destinées à remonter dans sentry.

#6 - 05 août 2022 14:02 - Benjamin Dauvergne

Thomas Noël a écrit :

Benjamin Dauvergne a écrit :

En fait on en manque un bon paquet (et il y a aussi tout ce qui est logger avec `logged_errors` qui ne remonte pas, mais ça c'est plus délicat).

C'est assez normal à mon sens : les `logged_errors` sont destinées aux administrateurs fonctionnels et ne sont donc pas destinées à remonter dans sentry.

On est HS (je demande juste à chopper toutes les Exception et pas juste celle de `psycopg2` au niveau du gestionnaire de requête) mais:

En théorie ce serait super, en pratique je ne suis pas certain qu'on soit exclusivement sur des erreurs de niveau fonctionnel "admin fonctionnel" et que dans ce cas le délais de traitement soit raisonnable, "l'admin fonctionnel le repère, il ouvre un ticket, etc...". Les traces qu'on reçoit via sentry/mail (voir le souci de date) on les traite plus vite en général (le cas `dmy/mdy` par exemple).

#7 - 05 août 2022 14:08 - Frédéric Péters

Il ne faut pas remonter les "logged errors" vers Sentry. Tout patch qui ferait ça serait rejeté.

#8 - 05 août 2022 15:03 - Nicolas Roche

- Fichier `0001-compat-add-foreground-Exception-to-sentry-67834.patch` ajouté

- Statut changé de En cours à Solution proposée

faut regarder le champ `mechanism`

(je ne l'ai pas trouvé)

je demande juste à chopper toutes les Exception et pas juste celle de `psycopg2` au niveau du gestionnaire de requête

J'ai fait ça (<https://sentry.entrouvert.org/entrouvert/publik/issues/62685/>)

#9 - 05 août 2022 15:13 - Frédéric Péters

compat:

Il y a peut-être la fausse idée comme quoi le préfixe d'un message de commit devrait correspondre au nom de fichier. Ce n'est pas le cas. "compat:" ça serait par exemple pour indiquer un commit qui touche à une question de compatibilité. Ça n'est pas le cas ici.

add foreground

tu as sans doute voulu écrire `frontoffice`.

~~

Sur le patch; le traitement des exceptions se fait actuellement dans `finish_failed_request`, voire `log_internal_error`, il ne faut pas avoir un bout pour sentry dans `compat.py` et un bout pour les mails ailleurs.

Pour boucler sur le début de mon commentaire : compat indique que ce fichier est une couche de compatibilité, entre django et quixote, il faut éviter d'y ajouter des fonctionnalités.

Alors oui il y a actuellement déjà

```
if can_sentry():  
    sentry_sdk.capture_exception(exc)
```

dans ce fichier pour une autre situation.

Mais celle-là aurait également été gérée dans `finish_failed_request()` ce ticket n'existerait même pas.

#10 - 05 août 2022 15:16 - Benjamin Dauvergne

Nicolas Roche a écrit :

faut regarder le champ `mechanism`

(je ne l'ai pas trouvé)

C'est uniquement sur les trace qui ne sont pas capturées par un mécanisme explicite de sentry (donc pas les trace dans les templates justement), ça en laisse quelques unes quand même.

#11 - 05 août 2022 15:20 - Benjamin Dauvergne

Frédéric Péters a écrit :

Alors oui il y a actuellement déjà

[...]

dans ce fichier pour une autre situation.

Mais celle-là aurait également été gérée dans `finish_failed_request()` ce ticket n'existerait même pas.

En plus clair je crois que Fred t'indique de travailler plutôt dans `finish_failed_request()` et de supprimer les références explicites à sentry ailleurs (l'exception est disponible dans `finish_failed_request` via `sys.exc_info`).

#12 - 05 août 2022 16:49 - Nicolas Roche

- Fichier `0001-misc-send-frontoffice-exceptions-to-sentry-too-67834.patch` ajouté

tu as sans doute voulu écrire `frontoffice`.

(misère)

#13 - 05 août 2022 16:52 - Frédéric Péters

Il faut faire ça après le

```
if exc_type is NotImplementedError:
```

basiquement, au niveau de `log_internal_error`.

#14 - 05 août 2022 17:12 - Nicolas Roche

- Fichier `0001-misc-send-frontoffice-exceptions-to-sentry-too-67834.patch` ajouté

#15 - 30 août 2022 14:16 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

#16 - 30 août 2022 15:29 - Frédéric Péters

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

commit b761fbf1345359d728cc15dc3f092105c7eb6310
Author: Nicolas ROCHE <nroche@entrouvert.com>

Date: Fri Aug 5 11:46:57 2022 +0200

misc: send frontoffice exceptions to sentry too (#67834)

#17 - 30 août 2022 16:14 - Transition automatique

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

#18 - 30 octobre 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0001-compat-add-foreground-transaction-error-to-sentry-67.patch	1,84 ko	05 août 2022	Nicolas Roche
0001-compat-add-foreground-Exception-to-sentry-67834.patch	1,19 ko	05 août 2022	Nicolas Roche
0001-misc-send-frontoffice-exceptions-to-sentry-too-67834.patch	4,01 ko	05 août 2022	Nicolas Roche
0001-misc-send-frontoffice-exceptions-to-sentry-too-67834.patch	4,25 ko	05 août 2022	Nicolas Roche