

w.c.s. - Bug #62164

crash sql sur filtre sur nombre interprété comme entier

25 février 2022 09:03 - Frédéric Péters

Statut:	Fermé	Début:	25 février 2022
Priorité:	Normal	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposé:	Oui		

Description

Exception:

```
type = '<class 'psycpg2.DataError'>', value = 'ERREUR: la valeur « 315610000204 » est en dehors des limites du type integer
```

```
...
clause = [<Equal (attribute: 'f92', value: 315610000011)>, <Equal (attribute: 'fbob7b6d448-99a7-464f-959b-12693ef5f5f5', value:
```

```
"L'Union")>, <Null (attribute: 'anonymised')>]
```

```
sql_statement = "SELECT id FROM formdata_25_collecte_des_encombrants_rende WHERE (CASE WHEN f92~E'^\\\\d+$' THEN f92::int ELSE NULL END) = %(c139790291423952)s AND fbob7b6d448_99a7_464f_959b_12693ef5f5f5 = %(c139790288052832)s AND anonymised IS NULL"
```

```
where_clauses = ["(CASE WHEN f92~E'^\\\\d+$' THEN f92::int ELSE NULL END) = %(c139790291423952)s", 'fbob7b6d448_99a7_464f_959b_12693ef5f5f5 = %(c139790288052832)s', 'anonymised IS NULL']
```

(le champ f92 est un champ liste, alimenté par une source de données jsonp).

sur les entiers on passe désormais par :

```
elif self.field and self.field.key in ['item', 'string'] and isinstance(self.value, int):
    # integer cast of db values
    attribute = "(CASE WHEN %s~E'^\\\\d+$' THEN %s::int ELSE NULL END)" % (attribute, attribute)
```

Et on a un entier parce que de manière large on tente le coup, wcs/backoffice/management.py

```
elif filter_field.type in ('item', 'items', 'string'):
    try:
        filter_field_value = int(filter_field_value)
    except ValueError:
        pass
```

Il me semble qu'on devrait ici tenter l'affaire uniquement sur item/items et si la source de données associée concerne des fiches.

Demandes liées:

Lié à w.c.s. - Development #60785: Form/Carddata API - filtre avec opérateur

Fermé

18 janvier 2022

Révisions associées

Révision 88534025 - 25 février 2022 13:58 - Frédéric Péters

backoffice: only cast "small" numbers to integers (#62164)

Révision 05c2aabb - 25 février 2022 13:59 - Frédéric Péters

backoffice: only cast "small" numbers to integers (#62164)

Historique

#1 - 25 février 2022 09:10 - Frédéric Péters

- Assigné à mis à Frédéric Péters

#2 - 25 février 2022 09:28 - Lauréline Guérin

- Lié à Development #60785: Form/Carddata API - filtre avec opérateur ajouté

#3 - 25 février 2022 09:29 - Lauréline Guérin

Note: conséquence de [#60785](#) et de filtrage avec opérateur (lt, lte, gt, gte, sur un champ string qui contient du numérique)

#4 - 25 février 2022 12:48 - Frédéric Péters

- Fichier 0001-backoffice-only-cast-small-numbers-to-integers-62164.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Voilà, la partie importante c'est le cast sql qui se fait uniquement si \d{1,9}, plutôt que \d+.

Parce que c'était la piste initiale et ça permet quand même de ne pas passer du tout par ce bout SQL, du côté python, condition pour taper uniquement en entier si ça tient sur 32 bits. (en passant la regex côté postgresql me fait penser que les nombres négatifs ne sont peut-être pas gérés, mais ça sera autre chose).

#5 - 25 février 2022 12:55 - Lauréline Guérin

- Statut changé de Solution proposée à Solution validée

#6 - 25 février 2022 13:59 - Frédéric Péters

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

```
commit 885340258f068c6a3a4d933598b2334267920897
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Fri Feb 25 09:25:16 2022 +0100
```

```
backoffice: only cast "small" numbers to integers (#62164)
```

(et je vais l'envoyer en hotfix aussi)

#7 - 26 février 2022 17:17 - Transition automatique

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

#8 - 01 mai 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0001-backoffice-only-cast-small-numbers-to-integers-62164.patch	5,17 ko	25 février 2022	Frédéric Péters
---	---------	-----------------	-----------------