

Authentic 2 - Bug #45256

utils: erreur de type sur fonction timestamp_from_datetime lorsque la zone horaire n'est pas précisée

18 juillet 2020 10:20 - Paul Marillonnet

Statut:	Fermé	Début:	18 juillet 2020
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

Parce qu'alors la méthode utcoffset de l'objet datetime renvoie None, et en python3 semblerait ça ne passe plus :

```
dt = datetime.datetime(2020, 7, 18, 10, 12, 41, 264101)
```

```
def timestamp_from_datetime(dt):
    '''Convert an aware datetime as an Unix timestamp'''
> utc_naive = dt.replace(tzinfo=None) - dt.utcoffset()
E TypeError: unsupported operand type(s) for -: 'datetime.datetime' and 'NoneType'
```

Révisions associées

Révision 9ce7263b - 18 juillet 2020 15:54 - Paul Marillonnet

utils: remove obsolete timestamp_from_datetime (#45256)

Historique

#1 - 18 juillet 2020 10:38 - Paul Marillonnet

- Fichier 0001-utils-do-not-assume-datetime-is-aware-when-creating-.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

#2 - 18 juillet 2020 11:44 - Benjamin Dauvergne

La méthode est inutile en python3, <https://docs.python.org/fr/3.6/library/datetime.html#datetime.datetime.timestamp> on peut la remplacer par dt.timestamp() sans souci.

#3 - 18 juillet 2020 16:18 - Paul Marillonnet

- Fichier 0001-utils-remove-obsolete-timestamp_from_datetime-45256.patch ajouté

Avec un cast nécessaire quand même car cette méthode renvoie un flottant.

Je viens de me rappeler que, contre toute intuition, int(toto) quand toto est un flottant ne prend pas l'entier le plus proche, juste la troncature.

Ça me paraît pas grave, mais si à la relecture on juge qu'une seconde de plus ou de moins ça peut être important pour le SSO, je referai le patch.

#4 - 18 juillet 2020 16:31 - Benjamin Dauvergne

Paul Marillonnet a écrit :

Avec un cast nécessaire quand même car cette méthode renvoie un flottant.

Je viens de me rappeler que, contre toute intuition, int(toto) quand toto est un flottant ne prend pas l'entier le plus proche, juste la troncature.

C'est quoi l'entier le plus proche de 1,5 ? La troncature c'est tout aussi bien que les milles autres façons arbitraires de le faire, déjà troncature t'en as au moins deux, vers -inf or vers 0. Et pour ta passion des arrondis : <https://docs.python.org/2/library/decimal.html#decimal.Context>

Ça me paraît pas grave, mais si à la relecture on juge qu'une seconde de plus ou de moins ça peut être important pour le SSO, je referai le patch.

Non ce n'est pas important.

#5 - 18 juillet 2020 16:31 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

#6 - 18 juillet 2020 16:39 - Paul Marillonnet

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

```
commit 9ce7263b817f9b4970dc20d7d4b6d39aeb28935e
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date: Sat Jul 18 10:35:29 2020 +0200
```

```
utils: remove obsolete timestamp_from_datetime (#45256)
```

#7 - 18 juillet 2020 16:49 - Paul Marillonnet

Benjamin Dauvergne a écrit :

C'est quoi l'entier le plus proche de 1,5 ? La troncature c'est tout aussi bien que les milles autres façons arbitraires de le faire, déjà troncature t'en as au moins deux, vers -inf or vers 0. Et pour ta passion des arrondis : <https://docs.python.org/2/library/decimal.html#decimal.Context>

On m'a toujours appris]0, 0.5[pour l'entier directement inférieur et [0.5, 1[va vers l'entier directement supérieur. Je croyais que c'était communément accepté, c'est tout un monde qui s'effondre :)

Ma crainte, sans doute infondée (et en tout cas pas justifiée ici), serait un cas d'usage, de calcul numérique ou autre, où on se trimballe un flottant y qui à un moment donné doit être interprété comme la représentation d'un entier $A - \epsilon$, où ϵ est négligeable, mais pas négligé par $\text{int}(y)$ qui renvoie donc $A - 1$, à tort.

#8 - 20 juillet 2020 16:15 - Frédéric Péters

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

Fichiers

0001-utils-do-not-assume-datetime-is-aware-when-creating-.patch	2,48 ko	18 juillet 2020	Paul Marillonnet
0001-utils-remove-obsolete-timestamp_from_datetime-45256.patch	10,8 ko	18 juillet 2020	Paul Marillonnet