

Passerelle - Development #51365

Connecteur eSirius

23 février 2021 12:01 - Nicolas Roche

Statut:	Fermé	Début:	23 février 2021
Priorité:	Normal	Echéance:	18 mars 2021
Assigné à:	Nicolas Roche	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

(dans un premier temps, pour étude cf <https://dev.entrouvert.org/issues/50051#note-23>)

- implémenter le token d'authentification décrit par <https://dev.entrouvert.org/issues/50051#note-8>
- taper sur `/appointments` pour implémenter :
ex:

```
$ curl -X GET "http://185.71.148.19/ePlanning/webservices/api/appointments/xxx" -H "accept: application/json; charset=utf-8"
{"code": "Not Found", "message": "Le rendez-vous {0} n'existe pas"}
```

- un endpoint d'enregistrement d'un rendez-vous
- un endpoint d'annulation d'un rendez-vous
- ~~un endpoint de modification d'un rendez-vous~~

Révisions associées

Révision 1481c7a6 - 08 mars 2021 10:59 - Nicolas Roche

esirius: add e-sirius connector (#51365)

Historique

#1 - 24 février 2021 09:47 - Nicolas Roche

- Assigné à mis à Nicolas Roche

#2 - 25 février 2021 18:50 - Nicolas Roche

- Description mis à jour

#3 - 25 février 2021 18:54 - Nicolas Roche

- Fichier 0001-esirius-add-e-sirius-connector-51365.patch ajouté

- Fichier workflow-esirius-delete.wcs ajouté

- Fichier workflow-esirius-create.wcs ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Ci-joint un workflow qui permet de tester la création (pour tester le rendu de la définition du schémas du corps de la requête POST me semble la partie la plus délicate).

Et tant qu'à faire également un workflow de suppression.

#5 - 26 février 2021 00:23 - Thomas Noël

Copyright (C) 2020 : raté, on est en 2021 ;)

Sépare avec une ligne vide les imports de Cryptodome et django, ça sera plus joli. Ajoute une ligne vide avant le début de la déclaration de CREATE_APPOINTMENT_SCHEMA pour faire un peu plus pep8.

Attention sur « 'needsConfirmation': {'type': 'boolean'}, », wcs ne sait pas envoyer de booléen facilement (sauf si ça correspond à une case à cocher dans le formulaire, ce dont je doute ici). Même chose pour les integer. Rappelle-toi que wcs ne sait envoyer que des strings : on voit dans ton test que CREATE_APPOINTMENT_PAYLOAD pourrait être délicat à construire par wcs.

« # error 500 if spaces are inserted » est un commentaire un peu inutile.

Pour la classe ESirius : dérive plutôt de HTTPResource qui va te faire gagner des méthodes d'accès HTTP gratuitement (même si y'en a pas à dans ce qu'on a testé, si ça se trouve y'en aura besoin ailleurs ou plus tard, genre un proxy ou un certif)

Ca s'écrit "eSirius" et pas "e-Sirius".

Appelle l'URL plutôt « ePlanning webservices URL » si jamais un jour on tape dans d'autres API eSirius et qu'il faudra indiquer d'autres URLs... Et mets un exemple avec https ;)

Il faut créer un self.request qui intègre le pre (ajout de l'auth) et le post (vérification du résultat). Ca fera des endpoints bien plus clairs. Intègres-y le calcul du token, il n'a rien à faire dans une fonction annexe qui ne sert jamais ailleurs.

Ce code là :

```
if not post_data.get('user'):
    post_data.update({'user': {}})
```

s'écrit plutôt :

```
if 'user' not in post_data:
    post_data['user'] = {}
```

Le check est bizarre, il fait appel à un webservice qui n'est pas utilisé dans le reste du connecteur... mais en regardant le swagger je vois que y'a rien d'autre, so let it be.

Lors du delete, sur le « response.status_code == 304 » : ça n'a pas de sens ici (304 = Not Modified). Il faut demander à eSirius si c'est le résultat normal, s'ils n'ont pas plutôt confondu avec un habituel 404.

Dans le swagger on voit qu'il existe un GET /appointments/{codeRDV} : ajoute-le dans les endpoints de ce connecteur, ça permettra (au moins) de vérifier ce qu'un POST a réussi à créer dans eSirius ; c'est important.

#6 - 26 février 2021 18:14 - Nicolas Roche

- Fichier workflow-esirius-delete(1).wcs ajouté
- Fichier workflow-esirius-create(1).wcs ajouté
- Fichier 0001-esirius-add-e-sirius-connector-51365.patch ajouté
- Echéance mis à 18 mars 2021

Remarques prises en compte ; je remets les 2 workflows de tests mis à jour.
Il reste encore à améliorer la gestions des erreurs (cf #51499 et #51500).

#7 - 27 février 2021 03:00 - Thomas Noël

Dans le code on parle de "booking" alors qu'il s'agit chez eSirius de "appointments" : gardons ce terme. En fait je propose d'utiliser "id" au lieu de "booking_id" (ça sera moins verbeux) et de dire "Appointment not found" lors du 304.

A ce propos, pour la gestion du delete/304, ajoute un petit commentaire "# handle strange 304 delete response".

delete_appointment : retourne data:None plutôt qu'un dico vide.

Dans request, le moment où tu appelles requests.get avec "params=payload" est moche, un payload c'est pas des paramètres ;) J'aurais mis deux arguments distincts « def request(self, uri, method='get', params=None, json=None): », aussi parce que c'est plus proche de ce que requests gère.

Aussi, lors de l'appel à cette self.request, précise method='post' ou method='delete' plutôt que juste 'post' ou 'delete', ça permet de mieux comprendre ce que c'est que ces 'post' et 'delete', je trouve.

Enfin, juste pour faire mon top-pénible : le %s de strftime ce n'est pas portable, c'est du CPython post 1999... allez, passons, it works ;)

Bref, pas grand chose !

#8 - 03 mars 2021 18:04 - Nicolas Roche

- Fichier 0001-esirius-add-e-sirius-connector-51365.patch ajouté

(remarques prises en compte).

Au cas où on souhaite pousser en recette un peu en avance,
en attendant d'avoir des réponses dans #51265 (j'ai forcé la clé à 8 octets minimum).

#9 - 03 mars 2021 18:48 - Thomas Noël

Sur `self.request`, le « `raise APIError('Method Not Allowed', http_status=405)` » : tu ne peux pas répondre ça au client qui t'appelle car ce n'est pas lui qui fait à fait l'erreur, c'est le code du connecteur. Il faudrait ici lever un `NotImplementedError`...

Mais inutile de s'encombrer, fait juste :

```
# creation des headers
...
response = self.requests.request(method=method, url=url, headers=headers, params=params, json=json)
if method == 'delete' and response.status_code == 304: # handle strange 304 delete response
    raise APIError('Appointment not found')
if response.status_code != 200:
    ...
```

Pour prévoir l'avenir et la possible disparition du token (sait on jamais vu les autres tickets), ne le génère que si `secret_key` existe.

Dans ton message de validation, ne dit pas "(longer keys are truncated)" puisque tu refuses les clés plus longues ;) Juste « DES key must be 8 bytes long »

Enfin pour "delete-appointment" je change d'avis (again) et te propose d'utiliser le verbe "delete". Parce que là on fait attends un POST avec un paramètre dans l'URL, sans payload, c'est tout bizarre, ça va être mal compris, éclaircissons. (et tant pis si pour l'instant la présentation n'est pas jolie dans le backoffice)

#10 - 03 mars 2021 18:49 - Benjamin Dauvergne

Nicolas Roche a écrit :

(remarques prises en compte).

Au cas où on souhaite pousser en recette un peu en avance, en attendant d'avoir des réponses dans #51265 (j'ai forcé la clé à 8 octets minimum).

On attend pas, les clés feront 8 octets et rien d'autre ne marchera, tu pourrais même mettre `min_length=8, max_length=8` sur le champ.

#11 - 03 mars 2021 18:53 - Thomas Noël

Benjamin Dauvergne a écrit :

On attend pas, les clés feront 8 octets et rien d'autre ne marchera, tu pourrais même mettre `min_length=8, max_length=8` sur le champ.

+1 ... et ça t'évite la validation à la main, c'est mieux. A voir si ça se cumule avec `blank=True`, qui nous permettrait de dire "c'est soit un token de 8 octets pile, soit pas de token".

#12 - 04 mars 2021 11:50 - Nicolas Roche

- Fichier `0001-esirius-add-e-sirius-connector-51365.patch` ajouté

Merci pour les relectures, j'ai laissé cependant quelques points en suspend...

fait juste : `response = self.requests.request(method=method, ...`

fait

token ..., ne le génère que si `secret_key` existe.

fait

pour "delete-appointment" ... utiliser le verbe "delete"

fait, mais du coup j'ai également adapté le endpoint 'get-appointment' à la sauce REST.

Dans ton message de validation, ne dit pas "(longer keys are truncated)" puisque tu refuses les clés plus longues

Bof, j'accepte aussi les clé plus longues : typiquement 'ES2I Info Caller Http Encryption Key', qui colle avec la clé fournie pas ESSI (sans que l'utilisateur ait à la tronquer pour taper 'ES2I Inf').

Je pense que c'est le message qui induit en erreur, il faudrait écrire quelque-chose comme : "les clé plus longues seront en définitive utilisées

tronquées à 8 octets". Mais qu'ici, on n'y gagne pas à simplifier le traitement pour simplifier le libellé (et non pas la saisie).

tu pourrais même mettre `min_length=8`

Bof, ça n'existe pas sur les modèles, d'où l'ergot pour la validation.

<https://docs.djangoproject.com/fr/2.2/ref/models/fields/#charfield>

cumule avec `blank=True`

oui, j'ai rajouté un test qui le met en évidence que c'est bien le cas.

Dernier point qui n'a pas été encore discuté.

Il me reste un soucis concernant #51500 : si l'on met une mauvaise clé (ou la bonne clé mais le mauvais "secret_id")

alors ESSI crash en 500 sur `get_status` et le connecteur est désactivé.

Ça me paraît un peu scabreux et du coup je traduis le crash Java comme quoi il y a bien quelqu'un qui me répond.

Voilà, je râle mais vos remarques feront loi en définitive.

#13 - 04 mars 2021 14:31 - Thomas Noël

Nicolas Roche a écrit :

pour "delete-appointment" ... utiliser le verbe "delete"

fait, mais du coup j'ai également adapté le endpoint 'get-appointment' à la sauce REST.

Nope nope nope. Pour le get comme pour le delete, laisse id en paramètre classique. C'est comme ça que ça se configure dans les appels webservices de wcs.

Dernier point qui n'a pas été encore discuté.

Il me reste un soucis concernant #51500 : si l'on met une mauvaise clé (ou la bonne clé mais le mauvais "secret_id")

alors ESSI crash en 500 sur `get_status` et le connecteur est désactivé.

Comment ça, désactivé ? Ils coupent l'accès ??

#14 - 04 mars 2021 16:04 - Nicolas Roche

ESSI crash en 500 sur `get_status` et le connecteur est désactivé.

Il s'agit de notre connecteur qui se met en stand-by, quand on met à jour les paramètres, je dirais ici :

```
class GenericCreateConnectorView(GenericConnectorMixin, CreateView):
    ...
    def form_valid(self, form):
    ...
        self.object.availability()
```

#15 - 04 mars 2021 16:38 - Thomas Noël

Nicolas Roche a écrit :

ESSI crash en 500 sur `get_status` et le connecteur est désactivé.

Il s'agit de notre connecteur qui se met en stand-by

Je ne comprends pas ce que tu dis. On n'a pas cette notion de "stand by". On a juste un truc qui dit "y'a un problème" et lance des alertes (mail) ; mais le connecteur est toujours là et utilisable.

Donc, pas de soucis, on aura des 500 en cas de pépin d'authentification... c'est comme ça. Ça ne devrait pas arriver souvent.

#16 - 04 mars 2021 17:06 - Nicolas Roche

- Fichier *Screenshot_2021-03-04 Passerelle.png* ajouté

On a juste un truc qui dit "y'a un problème" et lance des alertes (mail) ; mais le connecteur est toujours là et utilisable.

Oui : hors-service ne signifie pas en stand-by tu as raison
(Comme la "désactivation" fait que le connecteur arrête de loguer j'avais fini par croire que ça coupait l'accès).

Ce qui me gêne en fait ici (pour revenir à la raison initiale de ma remarque) c'est que le connecteur s'active/désactive (juste en visuel donc) en live en fonction de la validité du mot de passe tapé (mais j'avoue que je vois le verre à moitié vide alors que je peux aussi voir ça comme une super fixture).

Je voulais juste avertir et préciser #51500 (crash systématique sur mauvaise identification) et donc ok, je laisse passer les 500.

#17 - 04 mars 2021 17:11 - Frédéric Péters

Avec les mauvais paramètres de connexion le connecteur est hors service, ça me semble tout à fait normal.

#19 - 05 mars 2021 09:18 - Nicolas Roche

- Fichier 0001-esirius-add-e-sirius-connector-51365.patch ajouté

Remarques prise en compte :

- Pour le get comme pour le delete, laisse id en paramètre classique.
- pas de soucis, on aura des 500 en cas de pépin d'authentification.

#20 - 05 mars 2021 09:55 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

#21 - 08 mars 2021 11:00 - Nicolas Roche

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

```
commit 1481c7a6c3a455250103635c09a381675c443d7a
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Wed Feb 24 10:56:01 2021 +0100
```

```
esirius: add e-sirius connector (#51365)
```

#22 - 08 mars 2021 18:16 - Frédéric Péters

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

Fichiers

0001-esirius-add-e-sirius-connector-51365.patch	18,6 ko	25 février 2021	Nicolas Roche
workflow-esirius-create.wcs	3,42 ko	25 février 2021	Nicolas Roche
workflow-esirius-delete.wcs	2,11 ko	25 février 2021	Nicolas Roche
workflow-esirius-create(1).wcs	3,53 ko	26 février 2021	Nicolas Roche
workflow-esirius-delete(1).wcs	2,1 ko	26 février 2021	Nicolas Roche
0001-esirius-add-e-sirius-connector-51365.patch	22,9 ko	26 février 2021	Nicolas Roche
0001-esirius-add-e-sirius-connector-51365.patch	25,1 ko	03 mars 2021	Nicolas Roche
0001-esirius-add-e-sirius-connector-51365.patch	25 ko	04 mars 2021	Nicolas Roche
Screenshot_2021-03-04 Passerelle.png	10,5 ko	04 mars 2021	Nicolas Roche
0001-esirius-add-e-sirius-connector-51365.patch	24,8 ko	05 mars 2021	Nicolas Roche