Passerelle - Development #29829

connecteur système téléphonique générique

16 janvier 2019 17:48 - Thomas Noël

Statut:	Fermé	**Début:**	16 janvier 2019
Priorité:	Normal	**Echéance:**	
Assigné à:	Thomas Noël	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	

Description

Un connecteur dédié à recevoir des call-start et call-stop depuis un PABX lorsqu'un agent décroche ou raccroche son téléphone.

Le connecteur maintient les appels en cours.

Il dispose d'un endpoint permettant au portail usager de savoir :

- quels sont les appels en cours (et passés) sur le poste de l'agent
- quels sont les appels en cours (et passés) concernant un numéro d'appelant

Révisions associées

Révision 154e689b - 21 janvier 2019 17:31 - Thomas Noël

add phone calls connector (#29829)

Historique

#1 - 16 janvier 2019 17:50 - Thomas Noël

- Fichier 0001-add-phone-calls-connector-29829.patch ajouté

- Statut changé de Nouveau à En cours

- Patch proposed changé de Non à Oui

Pour info et éventuels premières remarques, le code qui fait le job ; restent les tests à écrire pour vérifier que tout ça marche bien.

#2 - 17 janvier 2019 11:38 - Thomas Noël

- Fichier 0001-add-phone-calls-connector-29829.patch ajouté

- Statut changé de En cours à Solution proposée

Voilà, avec les tests.

#3 - 21 janvier 2019 14:48 - Frédéric Péters

Plus tard, pour permettre au même agent de gérer plusieurs lignes, je pense que ça sera pratique d'avoir une API pour associer un nameid à une/plusieurs lignes, et que /calls sur réception d'un nameid retourne toutes les lignes concernées.

Mais en l'état, je pense que ça va marcher ainsi.

Seul truc, peut-être ne pas inclure la durée de l'appel dans la réponse à l'API, j'étais parti sur l'idée de comparer une nouvelle réponse reçue à la réponse précédente mais avoir duration qui bouge tout le temps empêche ça.

#4 - 21 janvier 2019 15:22 - Thomas Noël

- Fichier 0001-add-phone-calls-connector-29829.patch ajouté

Frédéric Péters a écrit :

> Plus tard, pour permettre au même agent de gérer plusieurs lignes, je pense que ça sera pratique d'avoir une API pour associer un nameid à une/plusieurs lignes, et que /calls sur réception d'un nameid retourne toutes les lignes concernées.
> Mais en l'état, je pense que ça va marcher ainsi.

Ok, je garde ça pour l'avenir.

> Seul truc, peut-être ne pas inclure la durée de l'appel dans la réponse à l'API, j'étais parti sur l'idée de comparer une nouvelle réponse reçue à la réponse précédente mais avoir duration qui bouge tout le temps empêche ça.

Yep, j'avoue que j'avais posé ça pour m'aider à faire mes tests avec mes yeux de pleins de clic-clic...

Interdiff :

```
diff --git a/passerelle/apps/phonecalls/models.py b/passerelle/apps/phonecalls/models.py
index 75783a1..80658a0 100644
--- a/passerelle/apps/phonecalls/models.py
+++ b/passerelle/apps/phonecalls/models.py
@@ -133,17 +133,14 @@ class Call(models.Model):
         if self.end_timestamp:
             is_current = False
             end_timestamp = make_naive(self.end_timestamp)
-            duration = (self.end_timestamp - self.start_timestamp).seconds
         else:
             is_current = True
             end_timestamp = None
-            duration = (now() - self.start_timestamp).seconds
         return {
             'caller': self.caller,
             'callee': self.callee,
             'start': make_naive(self.start_timestamp),
             'end': end_timestamp,
             'is_current': is_current,
-            'duration': duration,
             'details': self.details,
         }
diff --git a/tests/test_phonecalls.py b/tests/test_phonecalls.py
index 868c8ee..bc48cd8 100644
--- a/tests/test_phonecalls.py
+++ b/tests/test_phonecalls.py
@@ -60,7 +60,6 @@ def test_phonecalls_start_stop(app, phonecalls):
     assert 'start' in resp.json['data']
     assert resp.json['data']['end'] is None
     assert resp.json['data']['is_current'] is True
-    assert 'duration' in resp.json['data']
     assert resp.json['data']['details'] == {}
     assert Call.objects.count() == 1
     call = Call.objects.first()
@@ -147,7 +146,6 @@ def test_phonecalls_start_stop(app, phonecalls):
     assert resp.json['data'][0]['caller'] == '0687654321'
     assert resp.json['data'][0]['start'] is not None
     assert resp.json['data'][0]['end'] is not None
-    assert resp.json['data'][0]['duration'] is not None
     assert resp.json['data'][0]['is_current'] is False
     assert Call.objects.count() == 2
     assert Call.objects.filter(end_timestamp__isnull=True).count() == 1
```

#5 - 21 janvier 2019 16:09 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

Ok go ainsi.

#6 - 21 janvier 2019 17:32 - Thomas Noël

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

```
commit 154e689b3b1b7c63b2eaf2d6d61fa388b567d215
Author: Thomas NOEL <tnoel@entrouvert.com>
Date:   Wed Jan 16 15:36:13 2019 +0100

    add phone calls connector (#29829)
```

#7 - 22 janvier 2019 00:15 - Frédéric Péters

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

#8 - 12 avril 2019 10:27 - Benjamin Dauvergne

- Statut changé de Solution déployée à Fermé

Fichiers

0001-add-phone-calls-connector-29829.patch	10,4 ko	16 janvier 2019	Thomas Noël
0001-add-phone-calls-connector-29829.patch	20,7 ko	17 janvier 2019	Thomas Noël
0001-add-phone-calls-connector-29829.patch	20,4 ko	21 janvier 2019	Thomas Noël