# Hobo - Bug #7550

## redéploiement automatique en cas de changement d'un service

11 juin 2015 20:40 - Thomas Noël

| | | | |
|---|---|---|---|
| **Statut:** | En cours | **Début:** | 11 juin 2015 |
| **Priorité:** | Normal | **Echéance:** | |
| **Assigné à:** | | **% réalisé:** | 0% |
| **Catégorie:** | | **Temps estimé:** | 0:00 heure |
| **Version cible:** | | | |
| **Patch proposed:** | Oui | **Planning:** | |

**Description**

Soucis actuel:

- quand on ajoute un service, le temps du déploiement, ses metadonnées ne sont pas disponibles
- et donc le hobo-deploy d'authentic ne déploie par le nouveau SP
- et donc il faut systématiquement "redéployer à la main" quelques secondes après déploiement d'un SP.

On a (ou on aura) le même soucis de "redéployer quand ça change sans que hobo le demande" quand un provider qu'on ne contrôle pas changera ses métadonnées.

Il faudrait donc un système qui surveille régulièrement (chaque 5 min?) les métadonnées de la fédération et lance un message de déploiement en cas de modif.

Alternatives:

- quand on déploie un SP, lancer un reploiement quand il devient dispo
- avoir un bouton "redéployer maintenant" sur hobo (actuellement on peut mais c'est bidouille)
- rendre l'agent hobo capable d'informer le serveur hobo (message d'info via celery/amqp, genre "j'ai fini de déployé truc" ou "ce machin a bougé", à réflechir)

### Historique

**#1 - 12 juin 2015 03:29 - Frédéric Péters**

À partir du moment où un service est déclaré opérationnel, on pourait simplemnt relancer un déploiement général ?

**#2 - 12 juin 2015 08:16 - Benjamin Dauvergne**

Et si hobo était plus actif (donc faisait des checks réguliers de présence des services pas juste quand on regarde la page) et ne publiait un service qu'une fois celui-ci opérationnel (à chaque fois qu'il devient opérationnel en fait), ne serait-ce pas mieux ?

Les états possibles:

```
             (1)              (2)               (3)
Not-deployed -> operationnal <-> unavailable [ -> Un-deployed ]
```

- (1) ici on déploie
- (2) ici on devrait envoyer un mail
- (3) ici on devrait retirer le service

**#3 - 12 juin 2015 14:29 - Thomas Noël**

Benjamin Dauvergne a écrit :

> Et si hobo était plus actif (donc faisait des checks réguliers de présence des services pas juste quand on regarde la page) (...)

Je pense que oui, mais dans un cadre multi-tenant je ne vois pas encore bien comment faire cela.

**#4 - 12 juin 2015 16:18 - Benjamin Dauvergne**

Bah avec un démon. À noter que si on passer en mode pseudo-thread (gevent&co) on pourrait facilement lancer un thread "cron"; on pourrait ne passer que hobo en pseudo-thread (par sécurité avant de voir si ça marche bien).

#### #6 - 23 août 2015 19:42 - Frédéric Péters

*- Fichier 0001-agent-deploy-twice-in-case-of-new-services-7550.patch ajouté*

*- Statut changé de Nouveau à En cours*

*- Patch proposed changé de Non à Oui*

Voilà en patch une réponse au problème initial (authentic n'attrapant pas les metadata, parce que service pas encore up); c'est vraiment basique mais ça fait le job et ça permet à mon sens d'avancer. Le truc est simplement, quand un nouveau service est déployé (ce qui est détecté parce qu'il n'existait précédemment pas), un nouveau déploiement général est demandé, bam. (ça pourrait être optimisé de mille façon mais j'aimerais poser cette étape avant de réfléchir à des trucs plus complets ou complexes).

#### #7 - 24 août 2015 16:18 - Thomas Noël

Moins simple que je le pensais ; je croyais qu'on pouvait passer par une réponse, afin que le hobo (server) reste le seul émetteur... mais comme on broadcast, ben non, pas de réponse possible.

Donc, c'est un "ack" à mes yeux. Mais je verrais bien une relecture par un tiers.

(a part ça, "cmd_process.returncode != 1" serait sans doute bien remplacé par "cmd_process.returncode == 1" ?)

#### #8 - 24 août 2015 16:26 - Frédéric Péters

Tu veux sans doute dire "== 0" à la fin, et la réponse c'est que je teste explicitement sur le "== 1" pour ne pas prendre en compte la situation où la commande échoue pour une autre raison.

#### #9 - 24 août 2015 16:44 - Thomas Noël

Oui, je voulais dire ==0 (et je pensais que ça ne devait répondre "ça existe" que dans ce cas, mais c'est pas très important, les autres cas ne devraient pas se produire).

De fait, au niveau du "if exists:" dans hobo_deploy.py, on devrait raiser une exception "SystemError" dans un except: final général.

#### #10 - 24 août 2015 16:45 - Thomas Noël

Thomas Noël a écrit :

> ~~De fait, au niveau du "if exists:" dans hobo_deploy.py, on devrait raiser une exception "SystemError" dans un except: final général.~~

Je délire du n'importe quoi. Oubliez-moi pour aujourd'hui.

#### #11 - 24 août 2015 16:56 - Serghei Mihai (congés, retour 15/05)

Pourquoi executer new-site dans une queue séparée?

#### #12 - 24 août 2015 17:08 - Benjamin Dauvergne

Et pourquoi ne pas simplement relancer une tâche deploy depuis deploy ? Je me méfie des boucles mais là on peut en avoir une aussi; deploy -> new-site -> deploy ...

#### #13 - 24 août 2015 17:16 - Frédéric Péters

> Pourquoi executer new-site dans une queue séparée?

En passant du temps de configuration dans rabbitmq, ça permettrait de limiter le broadcast aux serveurs faisant tourner hobo.

> Et pourquoi ne pas simplement relancer une tâche deploy depuis deploy ?

C'était dans l'idée de rester dans un modèle où les deploy() sont lancés depuis le service hobo central mais ça pourrait être changé, je n'ai pas d'attachement particulier à ça.

#### #14 - 09 novembre 2015 15:05 - Benjamin Dauvergne

Bon on a pas mieux que ça et c'est tout ce qu'il manque pour un déploiement Montpellier automatique, donc ack. J'aurai bien pris des tests avec mais c'est pas évident.

#### #15 - 09 novembre 2015 15:40 - Frédéric Péters

Ok, je mettrai à jour le patch et le reproposerai.

#### #17 - 21 décembre 2016 12:53 - Frédéric Péters

En multi-publik ça devient encore plus sensible parce qu'il faut aller sur le hobo "parent" pour faire la modif bidon.

### Fichiers

| | | | |
|---|---|---|---|
| 0001-agent-deploy-twice-in-case-of-new-services-7550.patch | 6,13 ko | 23 août 2015 | Frédéric Péters |