# Dépanner les applications IOX qui ne peuvent pas démarrer ou qui s'arrêtent de manière inattendue

## Table des matières


Introduction

Informations générales

Informations de base sur la terminaison d'application IOX

Empêcher les conteneurs IOX de s'arrêter en cas de terminaison d'application/cible


## Introduction

Ce document décrit comment dépanner ou déboguer des applications IOX qui rencontrent un problème lors de leur démarrage ou qui s'arrêtent de manière inattendue.

## Informations générales

Lorsque vous développez des applications IOX, la meilleure approche consiste à le développer sur une autre plate-forme et/ou un sandbox. Une fois que l'application IOX souhaitée est testée et prête, elle peut être conditionnée et déployée sur un périphérique compatible IOX. Dans certains cas, ce déploiement ne fonctionne pas comme prévu, l'application peut s'arrêter de façon inattendue ou même ne pas démarrer.

Le comportement par défaut pour une machine virtuelle/un conteneur/une application IOX est de s'arrêter dès que la commande cible est terminée. Cela rend difficile le dépannage si quelque chose d'inattendu se produit, car toutes les informations non persistantes disparaîtront. Une autre conséquence est que cela prend beaucoup de temps lorsque vous voulez jouer/expérimenter avec les capacités des applications IOX. Pour démarrer une autre application/cible/commande/script, vous devez au moins modifier le package.yaml, créer le nouveau package, désactiver votre application actuelle, effectuer la mise à niveau avec le nouveau package, réactiver l'application et la démarrer.

## Informations de base sur la terminaison d'application IOX

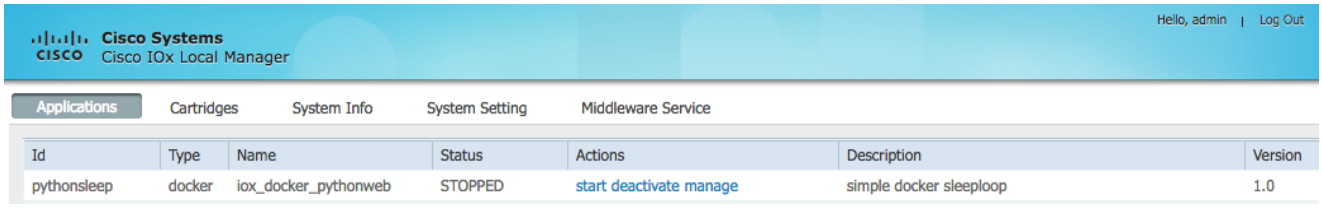
Local Manager et le client IOX ne fournissent pas beaucoup d'informations sur la raison pour laquelle les applications/conteneurs/machines virtuelles se sont arrêtés. Heureusement, le watchDog.log garde trace de cela et vous fournit également le dernier code de sortie/retour de l'application. Bien que cela ne puisse pas toujours vous aider, dans de nombreux cas, cela vous conduit à la raison que vous recherchez.

Pour récupérer le fichier watchDog.log avec le client IOX :

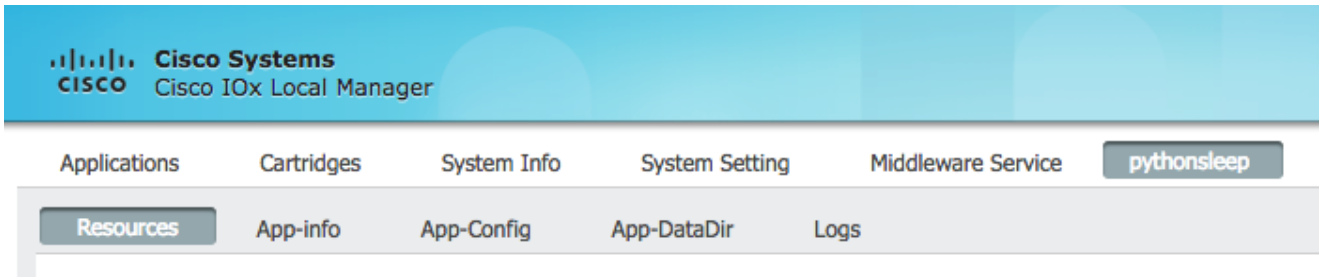
```
[jedepuyd@db ~]$ ioxclient app logs tail iox_docker_test watchDog.log 10
Currently active profile :  default
Command Name: application-logs-tail
App/Service : iox_docker_test, Logfile : watchDog.log, viewing last 10 lines
APP END TIME:1498207460
Time taken by App : 0 minutes and 0 seconds.
Got the ip address - 10.197.215.227 for interface eth0
All interfaces got the ips
APP START TIME:1498207536
App iox_docker_test started with PID : 11
Monitoring this process now
App iox_docker_test completed with exit code: 127
APP END TIME:1498207536
Time taken by App : 0 minutes and 0 seconds.
```

Pour extraire le journal à l'aide du Gestionnaire local :

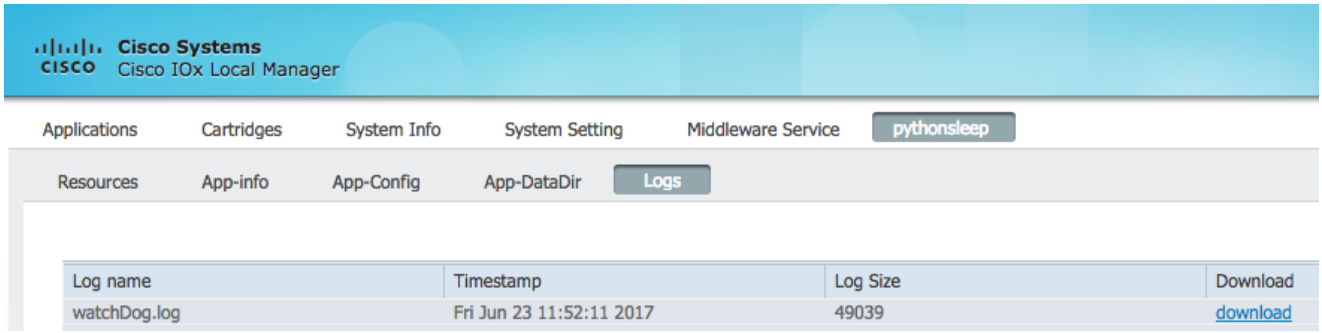
1. Connectez-vous au Gestionnaire local.
2. Comme l'illustre l'image, cliquez sur start deactivate manage pour l'application concernée




3. Sélectionnez l'onglet Journaux, comme illustré dans l'image :




4. Téléchargez le fichier watchDog.log




Par exemple, cette application est interrompue de façon inattendue et vous voyez ceci dans le

watchDog.log

```
APP START TIME:1498207536
App iox_docker_test started with PID : 11
Monitoring this process now
App iox_docker_test completed with exit code: 127
APP END TIME:1498207536
```

Comme vous pouvez le voir dans l'extrait de journal ci-dessus, le code de sortie était 127. Il s'agit d'un code de retour réservé et cela signifie : est introuvable, ce qui indique qu'une commande défectueuse est spécifiée en tant que cible ou que le script qui est démarré tente d'appeler une commande défectueuse.

Codes de sortie réservés les plus courants :

| RC | Signification | commentaire |
|---|---|---|
| 1 | Appel pour les erreurs générales | Erreurs diverses, telles que « division par zéro » et autres opérations non autorisées |
| 2 | Mauvaise utilisation des constructions en coque | Mot clé ou commande manquants, ou problème d'autorisation (et code de retour diff sur une comparaison de fichiers binaires ayant échoué). |
| 126 | Impossible d'exécuter la commande | La commande ou le problème d'autorisation n'est pas un exécutable |
| 127 | Commande introuvable | Problème possible avec $PATH ou une faute de frappe |
| 128 | Argument non valide à quitter (RC>255) | exit prend uniquement des arguments entiers compris entre 0 et 255 |
| 128+n | Signal d'erreur fatale "n" | le code de sortie renvoie 137 (128 + 9) -> le signal d'erreur de l'application était 9 |
| 130 | Terminé par Ctrl+c | Control-C est le signal d'erreur fatale 2, (130 = 128 + 2, voir ci-dessus) |
| 255 | État de sortie hors limites (RC>255) | exit prend uniquement des arguments entiers compris entre 0 et 255 |

Pour plus d'informations à ce sujet, consultez le site : http://tldp.org/LDP/abs/html/exitcodes.html

## Empêcher les conteneurs IOX de s'arrêter en cas de terminaison d'application/cible

La rubrique ci-dessus fournit des informations sur le dépannage d'une application défaillante, mais elle n'empêche pas l'arrêt de l'application IOX. Cela signifie que les informations les plus précieuses relatives au dépannage ont probablement disparu, car toutes les données non persistantes n'existent plus.

Comme mentionné ci-dessus, un autre exemple d'utilisation est de jouer avec les capacités de l'application IOX démarrée ou d'être flexible avec des commandes et des arguments.

Pour empêcher les applications IOX de se terminer à la fin de l'application, vous pouvez passer la commande -debug on à la commande activate :

```
[jedepuyd@db ~]$ ioxclient app activate -debug on testdebug
Currently active profile :  default
Command Name: application-activate
App testdebug is Activated
[jedepuyd@db ~]$ ioxclient app start testdebug
Currently active profile :  default
Command Name: application-start
App testdebug is Started
[jedepuyd@db ~]$ ioxclient app console testdebug
Currently active profile :  default
Command Name: application-console
Console setup is complete..
Running command : [ssh -p 2222 -i testdebug.pem appconsole@10.48.43.197]
/ #
```

Dans l'exemple ci-dessus, après avoir activé et commencé par l'indicateur -debug on, vous pouvez accéder au conteneur même si l'application est arrêtée. Vous pouvez lancer d'autres commandes ici et tester librement l'application dans l'environnement où elle s'exécute. Cela permet de gagner beaucoup de temps lors de la résolution de problèmes d'application ou de l'obtention de la bonne cible et des bons arguments.

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.