

IOX-toepassingen voor probleemoplossing die niet kunnen worden gestart of onverwacht zijn gestopt

Inhoud

[Inleiding](#)

[Achtergrondinformatie](#)

[Basisinformatie over IOX-toepassingsbeëindiging](#)

[Voorkomen dat IOX-containers ophouden bij toepassing/beoogde beëindiging](#)

Inleiding

Dit document beschrijft hoe u IOX-toepassingen kunt oplossen of oplossen die u onverwachts niet kunt starten of stoppen.

Achtergrondinformatie

Wanneer u IOX-toepassingen ontwikkelt, is de beste benadering om deze op een ander platform en/of zandbak te ontwikkelen. Nadat uw gewenste IOX-toepassing is getest en klaar is, kan deze worden verpakt en ingezet op een IOX-enabled-apparaat. In sommige gevallen werkt die inzet niet zoals verwacht, zou de toepassing onverwacht kunnen stoppen of zelfs niet kunnen starten.

Het standaardgedrag voor een IOX VM/container/applicatie moet stoppen zodra de doelopdracht is beëindigd. Dit maakt het moeilijk om een oplossing te vinden als er iets onverwachts gebeurt omdat alle niet-persistente informatie zal verdwijnen. Een ander gevolg is dat het zeer tijdrovend is als je met de mogelijkheden van IOX-toepassingen wilt spelen of experimenteren. Voor het starten van een andere app/target/opdracht/script moet u ten minste het pakket wijzigen.yaml, het nieuwe pakket bouwen, uw huidige toepassing deactiveren, upgraden met het nieuwe pakket, de toepassing opnieuw activeren en starten.

Basisinformatie over IOX-toepassingsbeëindiging

Plaatselijke Manager en IOX-client geven niet veel informatie over de reden waarom toepassingen/containers/VM's zijn gestopt. Gelukkig houdt het **horlogeDog.log** dit bij en voorziet u ook van de laatste exit/return code van de toepassing. Hoewel dit je niet altijd zal helpen, leidt het je in veel gevallen tot de reden waarom je op zoek bent.

Zo haalt u **horlogeDog.log** op bij IOX-client:

```
[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
```

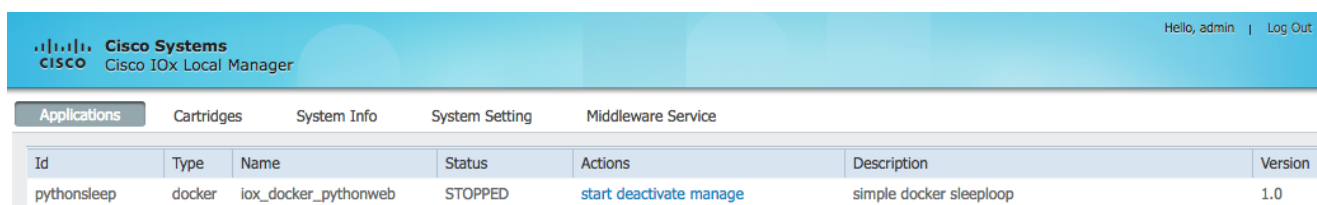
```

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.

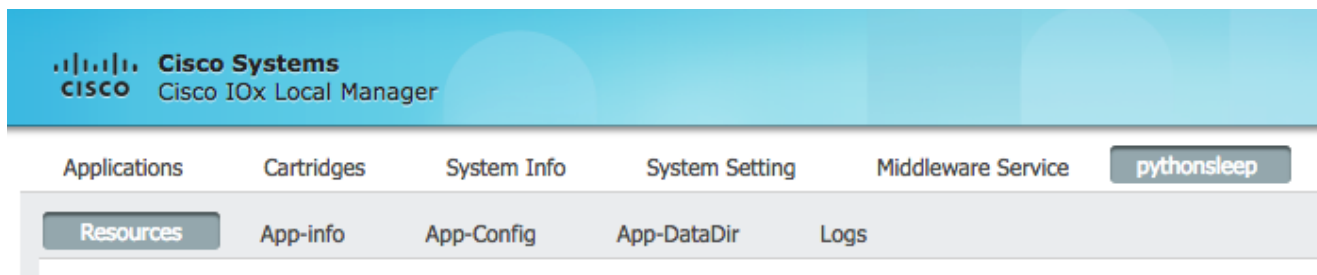
```

U kunt het logbestand als volgt ophalen via Local Manager:

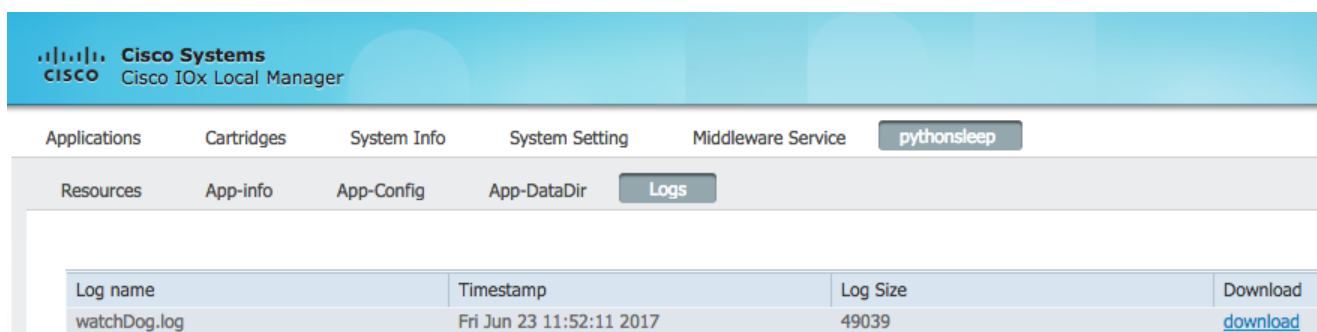
1. Meld u aan bij Local Manager.
2. Zoals in de afbeelding wordt aangegeven, klikt u op voor de desbetreffende toepassing op het deactiveren



3. Selecteer het tabblad **Logs**, zoals in de afbeelding:



4. Download het **horlogeDog.log**



Bijvoorbeeld, deze toepassing wordt onverwachts beëindigd en u ziet dit in **horlogeDog.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

```

Zoals je in het hierboven vermelde logfragment kunt zien, was de exit code 127. Dit is een **gereserveerde** return code die betekent: **Opdracht niet gevonden**, wat een defecte opdracht aangeeft wordt gespecificeerd als doel of het script dat wordt gestart probeert een foutieve

opdracht te bellen.

Meest voorkomende gereserveerde exit-codes:

RC	Betekenis	Opmerking
1	Catchall for general error	Diverse fouten, zoals "scheiden door nul" en andere ontoelaatbare verrichtingen
2	Misbruik van granaatgebouwen	Ontbrekende sleutelwoord of opdracht, of het probleem van de toestemming (en diff terugkeercodes op een mislukte binaire bestandsvergelijking).
126	Opdracht kan niet worden uitgevoerd	Het probleem of de opdracht van de vergunning zijn geen uitvoerbaar
127	Opdracht niet gevonden	Mogelijk probleem met \$PATH of een type
128	Ongeldig argument voor afsluiten (RC>255)	exit neemt alleen integer args in bereik 0 - 255
128+n	Fataal foutensignaal "n"	exit code retourneert 137 (128 + 9) -> app error signaal was 9
130	Beëindigd door Ctrl+c	Control-C is een fataal foutensignaal 2, (130 = 128 + 2, zie hierboven)
255	Status afsluiten buiten bereik (RC>255)	exit neemt alleen integer args in bereik 0 - 255

Meer informatie hierover is te vinden op: <http://tldp.org/LDP/abs/html/exitcodes.html>

Voorkomen dat IOX-containers ophouden bij toepassing/beoogde beëindiging

Het bovenstaande onderwerp geeft informatie over hoe u een falende toepassing kunt oplossen maar het voorkomt niet dat de IOX-toepassing stopt. Dit betekent dat de meest waarschijnlijke waardevolle informatie over de probleemoplossing is verdwenen omdat alle niet-persistente gegevens niet langer bestaan.

Zoals hierboven vermeld, is er een ander gebruiksgeschiedenis dat de mogelijkheden van de begonnen IOX-toepassing biedt, of dat hij flexibel is met opdrachten en argumenten.

Om te voorkomen dat IOX-toepassingen op het einde van de toepassing eindigen, kunt u het **-debug** doorgeven aan de activeringsopdracht:

```
[jdepuyd@db ~]$ ioxclient app activate -debug on testdebug
Currently active profile : default
Command Name: application-activate
App testdebug is Activated
[jdepuyd@db ~]$ ioxclient app start testdebug
Currently active profile : default
Command Name: application-start
App testdebug is Started
[jdepuyd@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]
/ #
```

In het bovenstaande voorbeeld, na het activeren en starten met het **-debug** op vlag, kunt u de container benaderen zelfs als de toepassing beëindigd is. U kunt hier andere opdrachten starten en vrij experimenteren met de toepassing in de omgeving waar uw applicatie draait. Dit bespaart veel tijd bij het oplossen van toepassingskwesaties of het krijgen van het juiste doel en de juiste

argumenten.