

排除IOX应用无法启动或意外停止故障

目录

[简介](#)

[背景信息](#)

[IOX应用终止的基本信息](#)

[防止IOX容器在应用/目标端接时停止](#)

简介

本文档介绍如何排除或调试遇到启动或意外停止问题的IOX应用。

背景信息

在开发IOX应用时，最好的方法是在另一个平台和/或沙盒上开发它。一旦您所需的IOX应用经过测试并准备就绪，即可将其打包并部署在支持IOX的设备上。在某些情况下，该部署不能按预期工作，应用可能会意外停止，甚至可能不会启动。

和IOX VM/container/application的默认行为是在目标命令终止后立即停止。这使得当发生意外情况时难以排除故障，因为所有非持久性信息都将消失。另一个结果是，当您想玩/尝试IOX应用的功能时，会非常耗时。启动另一个应用/target/command/script至少需要更改package.yaml、构建新包、停用当前应用、使用新包升级、重新激活应用并启动它。

IOX应用终止的基本信息

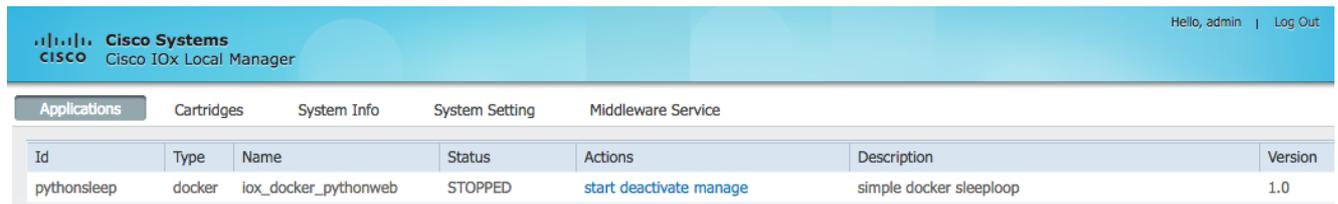
本地管理器和IOX客户端没有提供有关应用/容器/VM停止的原因的大量信息。幸运的是，**watchDog.log**会跟踪此信息，并向您提供应用程序的最后一个退出/返回代码。虽然这可能并不总是对您有帮助，但在很多情况下，它会引导您找到您所寻找的原因。

要使用IOX客户端获取watchDog.log:

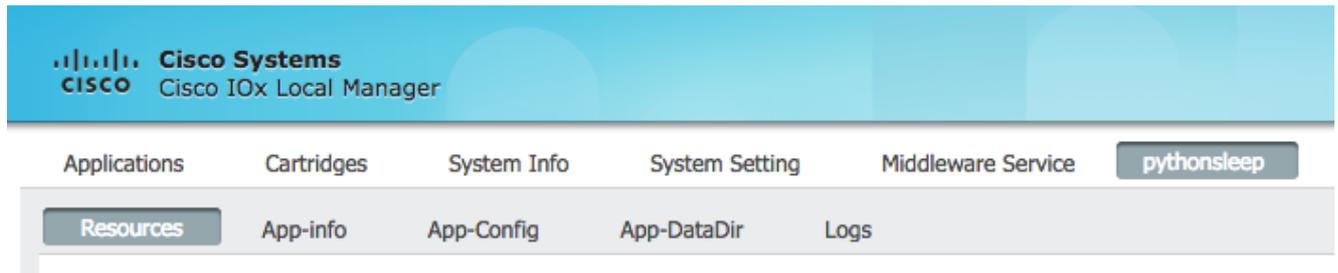
```
[jedepuyd@db ~]$ ioxclient app logs tail iox_docker_test watchDog.log 10
Currently active profile : default
Command Name: application-logs-tail
App/Service : iox_docker_test, Logfile : watchDog.log, viewing last 10 lines
APP END TIME:1498207460
Time taken by App : 0 minutes and 0 seconds.
Got the 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.
```

要使用本地管理器获取日志，请执行以下操作：

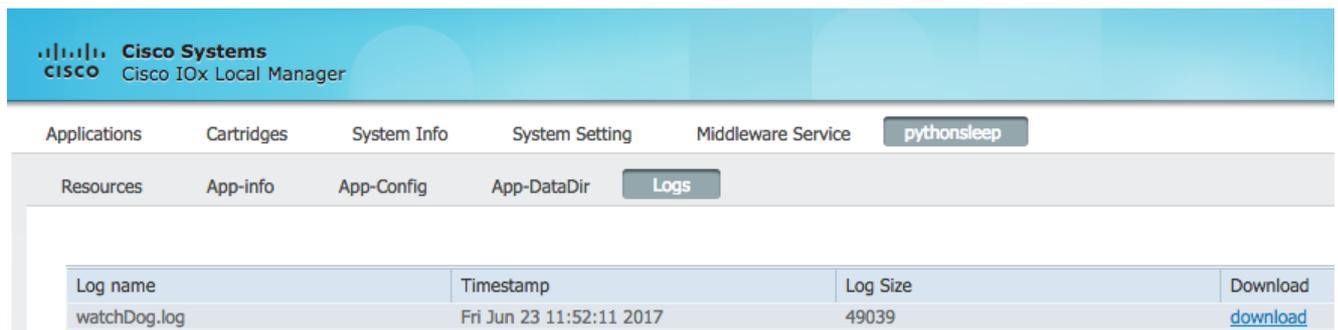
1. 登录到本地管理器。
2. 如图所示，单击开始停用**管理相关应用程序**



3. 选择“日志”选项卡，如图所示：



4. 下载watchDog.log



例如，此应用意外终止，您在watchDog.log中看到了这一点

```
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
```

如上面的日志片段所示，退出代码为127。这是保留的**返回代码**，表示：**命令未找到**，表示有故障的命令被指定为目标或启动的脚本尝试调用有故障的命令。

最常见的保留退出代码：

RC	含义	注释
1	捕捉一般错误	其他错误，如“除零”和其他不允许的操作
2	误用外壳构建	缺少关键字或命令，或权限问题（以及二进制文件比较失败时的差异返
126	无法执行命令	权限问题或命令不是可执行文件
127	找不到命令	\$PATH或拼写错误可能存在问题
128	要退出的参数无效(RC>255)	exit仅采用0 - 255范围内的整数参数
128+n	致命错误信号“n”	退出代码返回137(128 + 9)->应用错误信号为9
130	按Ctrl+c终止	Control-C是致命错误信号2，（130 = 128 + 2，请参阅上文）
255	退出状态超出范围(RC>255)	exit仅采用0 - 255范围内的整数参数

有关此项的详细信息，请访问：<http://tldp.org/LDP/abs/html/exitcodes.html>

防止IOX容器在应用/目标端接时停止

上述主题提供有关如何对故障应用进行故障排除的信息，但不会阻止IOX应用停止。这意味着，由于所有非持久数据不再存在，与故障排除相关的最可能有价值的信息已消失。

如上所述，另一个使用案例是使用已启动的IOX应用的功能或灵活使用命令和参数。

为防止IOX应用端终止，您可以将**-debug on**传递到activate命令：

```
[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]
/ #
```

在上例中，激活并以 `— debug on` 标志开头后，即使应用终止，您也可以访问容器。您可以在此处启动其他命令，并可以在应用程序运行的环境中自由地试验应用程序。这样可节省大量时间来解决应用程序问题或设置正确的目标和参数。