

CCM JAVA进程堆转储和碎片收集统计信息

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[问题](#)

[解决方案](#)

简介

本文如何描述添加Java虚拟机标志对在Cisco Call Manager (CCM)的JAVA_OPTS。

[先决条件](#)

[要求](#)

Cisco 建议您了解以下主题：

- Linux接口
- 虚拟机环境
- CCM组件

使用的组件

本文档中的信息基于以下软件版本：

- CloudCenter版本4.x
- CCM设备

背景信息

通常，TOMCAT进程去无答复和终止正常运行，但是没有指示原因的信息为什么TOMCAT是无答复。

[问题](#)

用户无法登录CloudCenter，并且登陆给CloudCenter的用户看到无答复CCM的用户界面(UI)。此问题可以被触发，因为TOMCAT服务不运作，并且停下来工作。然后，CCM默认日志不显示任何详细信息或原因至于TOMCAT服务为什么停下来工作。

解决方案

为了收集其他信息，在TOMCAT服务停下来运作前，JVM标志需要被添加到在 `/usr/local/tomcat/bin/setenv.sh` 的 `JAVA_OPTS`。这些参数帮助您收集堆沉默寡言JAVA进程和JAVA进程的碎片收集统计信息。然后，当TOMCAT进程变得无答复时，它生成 `/usr/local/tomcat/logs/ccm.bin` 和 `/usr/local/tomcat/logs/ccm-gc.log`。

步骤1.添加这些参数在 `/usr/local/tomcat/bin/setenv.sh`。

```
-XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/usr/local/tomcat/logs/ccm.bin -verbose:gc -
XX:+PrintGCDetails
-XX:+PrintGCDateStamps -Xloggc:/usr/local/tomcat/logs/ccm-gc.log
```

步骤2.导航对 `CCM > /usr/local/tomcat/bin/setenv.sh`

```
[root@ccm bin]# cat /usr/local/tomcat/bin/setenv.sh
export LD_LIBRARY_PATH=/usr/local/apr/lib
export CATALINA_PID=${CATALINA_HOME}/catalina.pid
export JPDA_ADDRESS=8000
export JAVA_OPTS="-Djava.awt.headless=true -Dfile.encoding=UTF-8 -server -Xms1024m -Xmx2048m -
XX:NewSize=512m
-XX:MaxNewSize=512m -XX:PermSize=512m -XX:MaxPermSize=512m -XX:+HeapDumpOnOutOfMemoryError
-XX:HeapDumpPath=/usr/local/tomcat/logs/ccm.bin -verbose:gc -XX:+PrintGCDetails -
XX:+PrintGCDateStamps
-Xloggc:/usr/local/tomcat/logs/ccm-gc.log -Dhttp.proxyHost=rcdn-california-proxy.cisco.com -
Dhttp.proxyPort=3128
-Dhttp.nonProxyHosts=10.201.210.1" export CLASSPATH=/usr/local/osmosix/etc export
CATALINA_OUT=/dev/null
```

步骤3.重新启动CCM TOMCAT进程。

```
/etc/init.d/tomcat restart
```

第四步：在重新启动，执行 `ps -ef` 命令后 `grep Tomcat` 为了确认TOMCAT进程使用被添加到 `/usr/local/tomcat/bin/setenv.sh` 的新的参数。

```
[root@ ccm ~]# ps -ef | grep tomcat
cliqrus+ 18683      1 99 16:46 ?          00:00:27 /usr/lib/jvm/java-8-sun/bin/java
-Djava.util.logging.config.file=/usr/local/tomcat/conf/logging.properties
-Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djava.awt.headless=true -
Dfile.encoding=UTF-8
-server -Xms1024m -Xmx2048m -XX:NewSize=512m -XX:MaxNewSize=512m -XX:PermSize=512m -
XX:MaxPermSize=512m
-XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/usr/local/tomcat/logs/ccm.bin -verbose:gc -
XX:+PrintGCDetails
-XX:+PrintGCDateStamps -Xloggc:/usr/local/tomcat/logs/ccm-gc.log -Dhttp.proxyHost=rcdn-california-proxy.cisco.com
-Dhttp.proxyPort=3128 -Dhttp.nonProxyHosts=10.201.210.1 -
Djava.endorsed.dirs=/usr/local/tomcat/endorsed
-classpath
/usr/local/osmosix/etc:/usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar
-Dcatalina.base=/usr/local/tomcat -Dcatalina.home=/usr/local/tomcat -
Djava.io.tmpdir=/usr/local/tomcat/temp
org.apache.catalina.startup.Bootstrap start root 18765 17947 0 16:47 pts/1 00:00:00 grep --
color=auto tomcat
```

这些新建的参数帮助您创建可以一次收集TOMCAT进程再去无答复的另外的文件：

- JAVA进程的堆转储- `/usr/local/tomcat/logs/ccm.bin`
- JAVA进程的碎片收集stats - `/usr/local/tomcat/logs/ccm-gc.log`