Foire aux questions de suite de virtualisation de données : Comment générez-vous un vidage mémoire de thread du processus de serveur CIS quand il ne répond pas ou le processus est arrêté ?

Contenu

Introduction

Ce document explique l'étape nécessaire pour générer les vidages mémoire (CIS) de thread de serveur d'informations de Cisco de suite de virtualisation de données sur des Plateformes de Microsoft Windows et d'Unix. Cisco les prennent en charge peut inviter un vidage mémoire CIS de thread afin d'examiner l'état du serveur si le serveur ne répond pas ou si un processus est arrêté.

Comment générez-vous un vidage mémoire de thread du processus de serveur CIS quand il ne répond pas ou le processus est arrêté ?

Utilisez les étapes fournies dans ce document pour générer cinq vidages mémoire de thread séparé pris à 30 intervalles seconde ou 1 minute.

Pour Windows

1. Mettez en marche le serveur composé à partir d'une session de cmd de Windows. Afin de faire ceci, entrez : **le <your installent le passage de dir> \ coffre \ composite_server.bat.Note**: La demande en session de cmd semble arrêtée, mais fonctionne toujours dans le premier plan.
2. Juste après que vous reproduisez la question de serveur, appuyez sur **Ctrl-Break** afin de générer un vidage mémoire de thread.

3. Pour visualiser le vidage mémoire de thread, allez au fichier nommé :
   `<install dir>\logs\cs_server.out.<yyyymmddhhmmss>`
   Par exemple, si vous génériez le vidage mémoire en avril 30 de thread, 2009 à 4:50 P.M., puis à vous visualiserait le fichier nommé :
   `cs_server.out.20120430165044`
   Note: Les vidages mémoire de fils multiple sont enregistré au même fichier.

Pour des unix et linux

1. La console d'Unix ou de Linux de commande, emploient une de deux méthodes pour obtenir l'ID de processus (PID) : Entrez : **picoseconde --E-F | Javas de grep** et recherchent le processus de Javas associé avec le serveur composé qui a été démarré à partir de votre répertoire d'installation.Exemple d'un processus de Javas :
   ```
   root 2079843812 9304 9294 0 0:02.28 ttys000 0:35.19 /usr/local/Composite_Software/
   CIS_5.2.0/jre/bin/java -server -XX:NewRatio=6 -XX:-UseGCOverheadLimit -XX:+HeapDump
   OnOutOfMemoryError -XX:HeapDumpPath=/usr/local/Composite_Software/CIS_5.2.0/logs
   -XX:PermSize=64m -XX:MaxPermSize=256m -Djava.endorsed.dirs=/usr/local/Composite_
   Software/CIS_5.2.0/apps/common/lib/endorsed -Dfile.encoding=UTF-8 -Dorg.apache.
   commons.logging.log.com.sun.xml.rpc=error -Dorg.apache.commons.logging.Log=org.
   apache.commons.logging.impl.Log4JLogger -Dlog4j.configuration=/usr/local/Composite
   _Software/CIS_5.2.0/conf/server/log4j.properties -Djava.security.properties=/
   usr/local/Composite_Software/CIS_5.2.0/conf/server/java.security -Dorg.mortbay.
   xml.XmlParser.Validating=false -Dorg.mortbay.jetty.servlet.SessionCookie=JSESSIONID
   :9400 -Dorg.apache.xml.dtm.DTMManager=com.compositesw.xml.dtm.pdtm.ProxyDTMMgrImpl
   -Dorg.apache.xml.utils.AbstractDOMBuilder=com.compositesw.xml.dtm.pdtm.ProxyDOM
   Builder -Xmx1024m -Dorg.apache.tuscany.sca.host.embedded.SCADomain=com.compositesw.
   server.soa.runtime.core.CisSCADomain -Dorg.apache.tuscany.sca.osgi.runtime.OSGi
   Runtime=com.compositesw.server.soa.runtime.core.CompositeOSGiRuntime -Dorg.osgi.
   service.http.port=9405 -Dapps.install.dir=/usr/local/Composite_Software/CIS_5.2.0
   -Dconf.install.dir=/usr/local/Composite_Software/CIS_5.2.0 -classpath /usr/local/
   Composite_Software/CIS_5.2.0/apps/base/lib/csbase.jar com.compositesw.base.boot.
   ServerBoot run
   ```
   Alternativement, vous pouvez sélectionner la commande : **Netstat --lnp | <port de grep que le composite est on> de écoute** afin de déterminer le PID.
2. Une fois que vous déterminez Java PID qui correspond à votre serveur, sélectionnez la commande : **mise à mort -3 <PID>**.
3. Le vidage mémoire de thread se produit dans un fichier nommé : le « <your installent le dir> \ logs \ cs_server.out.<yyyymmddhhmmss> ».Par exemple, un vidage mémoire de thread a généré en mars 22, 2011 à 3:37 P.M. serait nommé :
   `cs_server.out.20110322153729`
   Note: Les vidages mémoire de fils multiple sont enregistré au même fichier. Les Plateformes AIX génèreront une paire de fichiers de javacore* qui contiennent le vidage mémoire de thread et également un fichier de heapdump* dans le dir> de <install \ répertoire de logs.

Du studio (pris en charge en toutes les plateformes serveur)

1. Allez à la console de mémoire de gestionnaire de studio.
2. Cliquez sur la touche mémoire **inutilisée libre** pour vider un intervalle d'instantané dans « /logs/cs_server_status.log ». Ce log d'état contient légèrement abrégé, mais facilement accessible, vidage mémoire de thread.**Note**: Cette étape exige que votre serveur est assez sensible pour écrire au log d'état.
   Exemple d'un vidage mémoire de thread :
   `Full thread dump Java HotSpot(TM) Server VM (1.4.2_10-b03 mixed mode):`

```
"Timer Thread" daemon prio=2 tid=0x02ed0818 nid=0x288 waiting on condition
[580f000..580fd90] at java.lang.Thread.sleep(Native Method)at com.compositesw
.server.trigger.TimerCondition.run(TimerCondition.java:174) at java.lang.Thread
.run(Unknown Source)"Thread-32" prio=5 tid=0x03346d50 nid=0xb50 in Object.wait()
[57cf000..57cfd90] at java.lang.Object.wait(Native Method) - waiting on <0x104d0058>
(a java.util.ArrayList) at java.lang.Object.wait(Unknown Source) at com.compositesw.
server.services.structlog.LoggerManager$LogThread.run(LoggerManager.java:195) -
locked <0x104d0058> (a java.util.ArrayList)
```

Pour des Plateformes d'IBM AIX, le vidage mémoire de thread peut aller à un répertoire différent dans un fichier avec un nom semblable à 'javacore1306718.1174604495.txt. Dans cet exemple, l'emplacement du vidage mémoire de thread est trouvé dans le fichier cs_server.out dans le répertoire de /opt/Composite_Software/CIS3717/logs.

```
JVMDG217: Dump Handler is Processing Signal 3 - Please Wait.
JVMDG303: JVM Requesting Java core file
JVMDG304: Java core file written to /opt/Composite_Software/CIS_3.7.1/apps/
server/javacore1306718.1174604495.txt
JVMDG215: Dump Handler has Processed Dump Signal 3.
```