

Tidal Enterprise Scheduler: Ottenere un dump dello stack Java utilizzando "jstack"

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Convenzioni](#)

[Problema](#)

[Soluzione](#)

[Informazioni correlate](#)

[Introduzione](#)

In questo documento viene descritto come ottenere un dump dello stack Java utilizzando il comando **jstack** Java. Questo processo è valido su Windows, Linux e UNIX e su qualsiasi sistema Tidal che utilizza Oracle Sun Java versione 1.6+.

[Prerequisiti](#)

[Requisiti](#)

Cisco raccomanda la conoscenza di Java e del sistema operativo su cui viene eseguito Java. Inoltre, sul sistema deve essere installato Java JDK (Java Development Kit) versione 1.6+.

[Componenti usati](#)

- Sun Java JRE 1.6+
- Sun Java JDK 1.6+

[Convenzioni](#)

Per ulteriori informazioni sulle convenzioni usate, consultare il documento [Cisco sulle convenzioni nei suggerimenti tecnici](#).

[Problema](#)

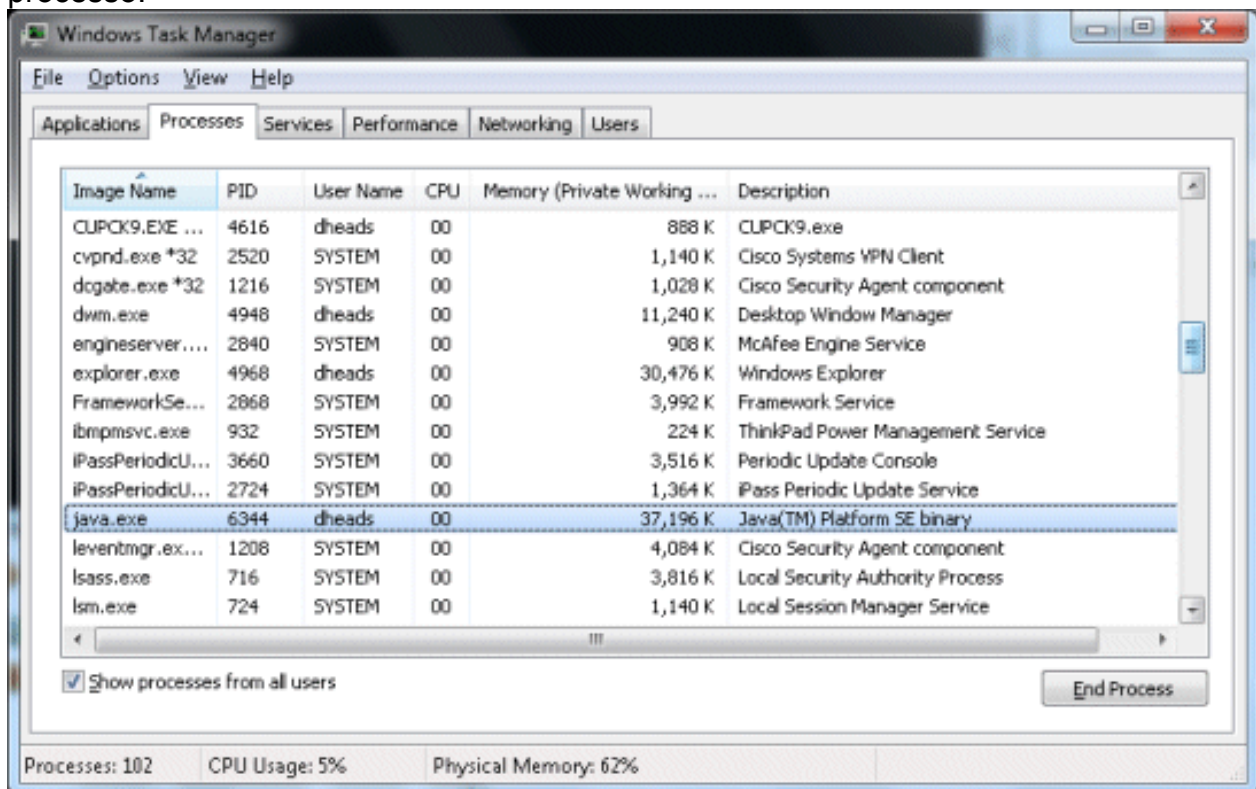
In alcuni casi, il supporto deve acquisire dati di thread di basso livello non presenti nei log master.

Soluzione

Per acquisire un dump dello stack Java utilizzando lo strumento **jstack** fornito da Oracle Sun, attenersi alla seguente procedura.

Nota: il processo Java deve eseguire Sun Java 1.6+ per eseguire il comando **jstack** e JDK deve essere installato sul sistema. Scaricare Oracle Sun JDK [qui](#) .

1. Individuare il numero di identificazione del processo sul computer su cui è in esecuzione il processo Java. Su un computer Windows, è possibile utilizzare Task Manager per trovare il numero di identificazione del processo.



In particolare, il **PID** è il numero di identificazione del processo. Se questa colonna non viene visualizzata, selezionare **Visualizza > Colonne > PID (ID processo)** per aggiungerla. In UNIX/Linux è possibile utilizzare il comando **ps -ef | grep UnixMaster** per ottenere l'ID processo:

```
root@sunts06:/# ps -ef | grep UnixMaster
root 5452 5423 0 14:43:21 pts/14 0:00 grep UnixMaster
root 1868 1 0 Jun 09 ? 31:49 /usr/bin/java -DN=UnixMaster -DTI
DAL_HOME=/opt/TIDAL/master/bin/.. -Xms256m -Xm
root@sunts06:/#
```

2. Una volta ottenuto il numero di identificazione del processo necessario, andare alla directory **BIN** in cui è installato Sun JDK ed eseguire il comando **jstack {PID}**, dove **{PID}** è il numero di identificazione del processo. Le informazioni dello stack Java verranno stampate sullo schermo.

```
C:\Program Files\Java\jdk1.6.0_25\bin>jstack 3724
2011-06-30 20:07:37
```

```
Full thread dump Java HotSpot(TM) 64-Bit Server VM (20.0-b11 mixed mode):
```

```
"Swing-Shell" daemon prio=6 tid=0x0000000006cce800 nid=0x1bd8 waiting on conditi
on [0x000000000843f000]
```

```
java.lang.Thread.State:WAITING (parking)
  at sun.misc.Unsafe.park(Native Method)
  - parking to wait for <0x00000000fb38ace8> (a java.util.concurrent.lock
s.AbstractQueuedSynchronizer$ConditionObject)
  at java.util.concurrent.locks.LockSupport.park(Unknown Source)
  at java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject
.await(Unknown Source)
  at java.util.concurrent.LinkedBlockingQueue.take(Unknown Source)
  at java.util.concurrent.ThreadPoolExecutor.getTask(Unknown Source)
  at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)
  at sun.awt.shell.Win32ShellFolderManager2$ComInvoker$3.run(Unknown
Source)
  at java.lang.Thread.run(Unknown Source)
```

```
"Thread-2" prio=6 tid=0x0000000006cd0800 nid=0xe6c waiting on condition [0x00000
000833f000]
```

```
java.lang.Thread.State: TIMED_WAITING (sleeping)
  at java.lang.Thread.sleep(Native Method)
```

3. Per reindirizzare le informazioni dello stack su un file, usare lo stesso comando con l'opzione di creazione del file che funziona sia in Windows che in Linux/UNIX:

```
jstack {PID} > [filename.out]
```

[Informazioni correlate](#)

- [Documentazione e supporto tecnico – Cisco Systems](#)