

Tidal Enterprise Scheduler: Get a Java Stack Dump Using "jstack"

Document ID: 113122

Contents

Introduction

Prerequisites

Requirements

Components Used

Conventions

Problem

Solution

Related Information

Introduction

This document describes how to get a Java stack dump using the Java **jstack** command. This process is valid on Windows, Linux, and UNIX, and any Tidal systems using Oracle Sun Java version 1.6+.

Prerequisites

Requirements

Cisco recommends that you have Java knowledge and the knowledge of the operating system on which Java runs. Also, the Java JDK (Java Development Kit) version 1.6+ will need to be installed on the system.

Components Used

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

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Problem

Occasionally, support needs to capture low-level thread data which is not in the master logs.

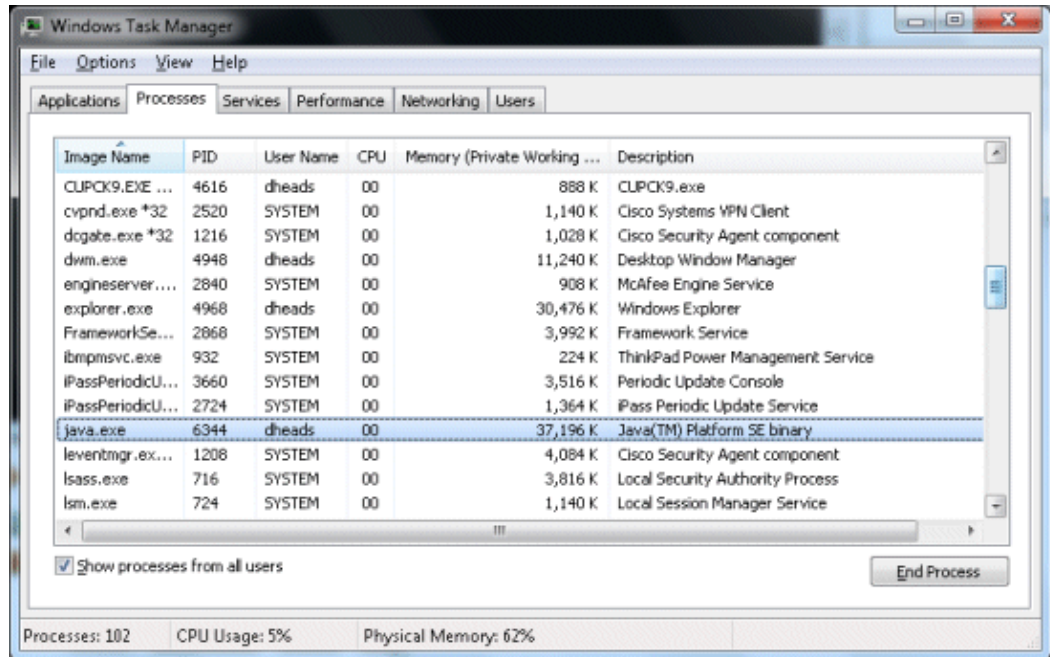
Solution

In order to capture a Java stack dump using the **jstack** tool provided by Oracle Sun, complete these steps.

Note: The Java process must be running Sun Java 1.6+ in order to run the **jstack** command, and the JDK must be installed on the system. Download the Oracle Sun JDK here [🔗](#).

1. Find the process identification number on the machine running the Java process.

- ◆ On a Windows machine, you can use the Task Manager to find the process identification number.



Specifically, the **PID** is the process identification number. If you do not see this column, select **View > Columns > PID (process identifier)** in order to add it.

- ◆ On UNIX/Linux, you can issue the `ps -ef | grep UnixMaster` command in order to get the process ID:

```
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. Once you have the process identification number you need, go to the **BIN** directory where the Sun JDK is installed and issue the `jstack {PID}` command where `{PID}` is the process identification number. This will print the Java stack information to the screen.

```
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=0x000000006cce800 nid=0x1bd8 waiting on condition
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=0x000000006cd0800 nid=0xe6c waiting on condition [0x0000
0000833f000]
```

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

3. In order to redirect the stack information to a file, issue the same command with the file creation option which will function in both Windows and Linux/UNIX:

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

Related Information

- **Technical Support & Documentation – Cisco Systems**
-

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2014 – 2015 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Jul 26, 2011

Document ID: 113122
