

Re: Terminating a process tree

Source: <http://coding.derkeiler.com/Archive/Java/comp.lang.java.programmer/2004-11/0947.html>

From: Wayne Marrison (wayne_at_wmcomputing.com)

Date: 11/09/04

Date: Tue, 9 Nov 2004 11:34:46 -0000

"marcus" <marcus@welby.md> wrote in message
news:41902F06.1010001@welby.md...

> *Hey Wayne,*
> *Since you agree this isn't the correct way to do it and you need a hack, I*
> *can suggest one that has worked for me in FreeBSD (should work in any*
> *linux type environment). Instead of launching the second java program*
> *directly I used a control script (stolen from my apache setup, I think)*
> *that records the PID in a dedicated text file. When I needed to kill it I*
> *invoked the control script, which issued the OS kill command on the*
> *recorded PID. Kills it dead as far as I know.*

>
>

> *Wayne Marrison wrote:*
>> *(posted yesterday in comp.lang.java.help, but not sure if that was the*
>> *right place and had no response, so posting in here today, sorry for the*
>> *additional bandwidth to those that subscribe to both groups)*

>>

>> *Hi group,*
>>
>> *I'm new here, so apologies if I ask this in the wrong way.*

>>

>> *I have been programming in java for about 2 months, and am currently*
>> *trying*
>> *to control an external application that I start using:*

>>

>> *Process proc;*
>> *Runtime rt;*
>>
>> *proc = rt.exec("java.exe -cp ... etc.. something");*

>>

>> *Then I capture the input/output & error streams and thread them off.*

>>

>> *The problem I have, is that I need to architect for the eventuality of*
>> *the*

>> *external process locking up and not responding to requests. I have the*
>> *whole capturing of streams in a timeout, so I can tell when the external*
>> *program has not responded within a reasonable time, however when I issue*

>> *the*
>> *proc.destroy() method, the java.exe process takes up almost 100% cpu and*
>> *never stops. I have to use operating system tools to perform a manual*
>> *kill*
>> *of the process.*
>>
>> *I can only assume that the proc.destroy() doesnt kill the entire process*
>> *tree, and because the command line starts off java.exe, which runs the*
>> *something.class file, a process is orphaned somewhere.*
>>
>> *Any help would be much appreciated.*
>>
>> *Thanks*
>>
>> *Wayne*
>>
>>
>

Hi Marcus,

I'm not sure how I would achieve this under Windows. I have looked at calling the "kill.exe" or the "taskkill.exe" directly, but then fall into the trap of trying to get the PID of the command I have just started. I know there are suggestions on using JNI to achieve that, but the learning curve is too great for me at this point.

I dont know how under windows to get the PID of an application using batch/script commands .. maybe I'd best rethink this as the complexities are starting to outweigh the benefits.

Thanks for your help.

Wayne