
Subject: Re: Write an app to start and kill another app periodically

Posted by [koldo](#) on Thu, 29 Apr 2010 15:51:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

jpdercyck wrote on Thu, 29 April 2010 17:33Hello,
Under Linux, I would like to write a small application which can check for a hung application periodically. If it is hung, it should kill the process and restart the application.

The application to be checked periodically was written in u++: it reads a data module acquisition connected by usb every 10 seconds. It runs for +/- 2 months and then it crash. I suppose the problem is with the usb driver or the usb module itself. I made the same program reading a different module connected by rs232 port with no problem.

I know that I have to use command like 'killall app'
How to start this kind of command from an u++ gui program ?

The best should be if the small checking app was a linux service. I saw a service app sample for windows, but does anybody have sample for linux ?

best regards
Jean-Paul

Hello jpdercyck

I am not an expert but this works.

You can use function `int kill(pid_t pid, int sig);`

pid is the id of the process to be killed.

There are these options from less to more aggressive:

```
#include <sys/types.h>
#include <signal.h>
#include <unistd.h>
...
pid_t pid;
...
kill(pid, SIGTSTP);
kill(pid, SIGTERM);
kill(pid, SIGHUP);
kill(pid, SIGKILL);
```

You can put `sleep()` between `kill()` functions just to give the program a little opportunity to stop itself.

To get the pid you can use `getpid()` and put this value somewhere where the killer process can take it.

