

Re: How to capture pid

Source: <http://coding.derkeiler.com/Archive/Perl/perl.beginners/2004-03/0180.html>

From: Zentara (zentara_at_highstream.net)

Date: 03/03/04

To: beginners@perl.org

Date: Wed, 03 Mar 2004 09:14:29 -0500

On Tue, 02 Mar 2004 18:07:58 -0600, reader@newsguy.com (Harry Putnam)

wrote:

>

>*For some reason the pid printed does not match the output of ps:*

>

>*actual sample script"*

>

>*Note the script outputs 15173 and ps shows 30335*

>

>*Do you have an idea what is happening here?*

Yeah, you are getting the parent pid in the perl script.

When you try to use system or exec to start another process, perl gets the pid of the very first part of the whole transaction, which is usually the pid of the xterm or the bash shell which will run your command with a second pid. It usually works ok, because when you kill the parent pid, it's children get killed too, so we hardly ever have to deal with them being different.

Use Proc::ProcessTable to get the pid of your program name.

Here is an example:

```
#!/usr/bin/perl
use warnings;
use strict;
use Proc::ProcessTable;
my $time = time();
my $xtermparentpid;
my $pttydev;

system("xterm -T $time &");
#specify title so you can match for it

my $t = new Proc::ProcessTable;
foreach my $p (@{$t->table}) {
    if($p->cmdline =~ /$time/){$xtermparentpid = $p->pid}
```

perl.beginners: Re: How to capture pid

```
}  
foreach my $p (@{$t->table}) {  
    if($p->ppid == $xtermparentpid){$ptydev = $p->ttydev}  
    }  
  
open(FH,">$ptydev") or warn $!;  
for(1..1000){print STDOUT $_, "\n"; print FH $_. 'a', "\n"; sleep 1;}  
  
close FH;  
  
--  
I'm not really a human, but I play one on earth.  
http://zentara.net/japh.html
```