
Subject: Re: idl process id

Posted by [Vap User](#) on Tue, 21 Apr 1998 07:00:00 GMT

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

korpela@islay.ssl.berkeley.edu (Eric J. Korpela) writes:

```
>
> In article <6hgr8e$qs3$1@nnrp1.dejanews.com>, <csaute3@alumni.umbc.edu> wrote:
>> I don't want to do "ps | grep idl" because of the result.
>>
>> IDL> spawn,'ps | grep idl',result,/sh
>> IDL> print, result
>> 23847 pd S    0:02 /usr/local/rsi/idl_4/bin/bin.sunos.4.1/idl
>> 23852 pd S    0:00 sh -c ps | grep idl 23853 pd S    0:00 grep idl
>
> Try this:
>
> IDL> spawn,'ps | grep idl | grep -v grep',result,/sh
> IDL> print,result
> 23847 pd S    0:02 /usr/local/rsi/idl_4/bin/bin.sunos.4.1/idl
> IDL> print,long(result)
>      23847
>
> If you really feel lucky you can try something like.....
>
> IDL> ; check your libc version before doing this!
> IDL> result=call_external('/lib/libc.so.1.9','getpid')
>      ~~~~~~ ; Great stuff!
> IDL> print,result
>      23847
>
```

Eric, this is truly amazing. One of the best tips I've seen! Thanks!

I don't speak for JPL, it doesn't speak for me.

Well, not all the time, at least.

William Daffer <vapuser@haifung.jpl.nasa.gov>

Subject: Re: idl process id

Posted by [L. Paul Mix](#) on Tue, 21 Apr 1998 07:00:00 GMT

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

csaute3@alumni.umbc.edu wrote:

```
>
> This is my setup:

> How would I easily grab the process ID of my IDL session?
```

>
> I don't want to do "ps | grep idl" because of the result.
>
> IDL> spawn,'ps | grep idl',result,/sh
> IDL> print, result
> 23847 pd S 0:02 /usr/local/rsi/idl_4/bin/bin.sunos.4.1/idl
> 23852 pd S 0:00 sh -c ps | grep idl 23853 pd S 0:00 grep idl
>
> There is an option on spawn to return the process ID:
>
> IDL> spawn,'ps | grep idl',result,/sh,pid=processid
> IDL> help, processid
> PROCESSID LONG = 23852
>
> But pid is for the child process. I would like the process id for
> the "parent process."
>
> Cathy Campo
>
> -----== Posted via Deja News, The Leader in Internet Discussion ==-----
> <http://www.dejanews.com/> Now offering spam-free web-based newsreading

Try:

```
spawn, 'echo $$', pid  
print, pid(0)
```

This will not work with Windows or MAC.

Paul Mix

Subject: Re: idl process id

Posted by [korpela](#) on Tue, 21 Apr 1998 07:00:00 GMT

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

In article <6hgr8e\$qs3\$1@nnrp1.dejanews.com>, <csaute3@alumni.umbc.edu> wrote:

> I don't want to do "ps | grep idl" because of the result.
>
> IDL> spawn,'ps | grep idl',result,/sh
> IDL> print, result
> 23847 pd S 0:02 /usr/local/rsi/idl_4/bin/bin.sunos.4.1/idl
> 23852 pd S 0:00 sh -c ps | grep idl 23853 pd S 0:00 grep idl

Try this:

```
IDL> spawn,'ps | grep idl | grep -v grep',result,/sh  
IDL> print,result  
23847 pd S 0:02 /usr/local/rsi/idl_4/bin/bin.sunos.4.1/idl
```



```
>>
> Try:
> spawn, 'echo $$', pid
> print, pid(0)
```

This returns the pid of the child process, which is exactly what she didn't want. The more I think about it, the more I like "pid=call_external('/lib/libc.so.1.9','getpid')"

Eric

--

Eric Korpela | An object at rest can never be
korpela@ssl.berkeley.edu | stopped.
Click for home page.

Subject: Re: idl process id
Posted by [steinhh](#) on Thu, 23 Apr 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

```
>> IDL> result=call_external('/lib/libc.so.1.9','getpid')

> Does anyone know how to get this to work on a DEC alpha?
> I can't find a shared object version of libc in my /lib
> directory.
```

Look in /shlib instead:

```
IDL> result=call_external('/shlib/libc.so','getpid')
IDL> print,result
    31185
```

With the easy-going way of adding new system routines that I found two days ago, just compile the below included C-code into a shareable object file (remember to link in \$IDL_DIR/bin/bin.alpha), initialize with

```
IDL> print,call_external('gmain.so','addmain')
```

and then you go:

```
IDL> print,getpid()
```

(This is perhaps not a good example of this utility - since it's possible to do this with a single-line call_external statement, but it does illustrate how easy it is to expand IDL by adding "built-in" routines that are able to return honest-to-RSI *variables*).

Regards,

Stein Vidar

-----gmain.c

```
#include <unistd.h>
#include <stdio.h>
#include "export.h"

#define NULL_VPTR ((IDL_VPTR) NULL)

IDL_VPTR GETPID(int argc, IDL_VPTR argv[])
{
    IDL_VPTR tmp = IDL_Gettmp();

    if (tmp==NULL_VPTR) {
        IDL_Message(IDL_M_NAMED_GENERIC,IDL_MSG_LONGJMP,
"Couldn't create temporary variable");
    }

    tmp->type = IDL_TYP_LONG;
    tmp->value.l = getpid();

    return tmp;
}

IDL_SYSFUN_DEF main_def[] = {{{(IDL_FUN_RET) GETPID,"GETPID", 0, 0}};

IDL_LONG addmain(int argc,char *argv[])
{
    IDL_AddSystemRoutine(main_def,IDL_TRUE,1); /* Just add getpid */
    return 0;
}
```
