

## Re: ptrace and thread

**Source:** <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2004-01/0062.html>

---

**From:** Julian Elischer ([julian\\_at\\_elischer.org](mailto:julian_at_elischer.org))

**Date:** 01/14/04

Date: Wed, 14 Jan 2004 13:34:02 -0800 (PST)

To: David Xu <[davidxu@freebsd.org](mailto:davidxu@freebsd.org)>

On Wed, 14 Jan 2004, David Xu wrote:

> *I am current working on debug support for KSE thread program, however I  
> found  
> ptrace interface is not thread-aware, in a threaded program, I need to  
> get/set registers  
> set for individual threads, current ptrace can not support that  
> features, there are two  
> ways to support these requirements:*

Yes I tried to address this a bit around the time when I added the single treading code. Ialso made several posts looking for advice from gdb/ptrace experts but got very little response..

As you noticed, the ptrace facility is almost completely useless WRT threads..

it is possible to imagine an extension where you select a single thread of interest but you would have to decide whether you want all the other threads to be left running or left suspended..

(you may need both possibilities to correctly debug a problem)

The problem is that the thread becomes invisible to the kernel when it crosses over to userland so the UTS needs to take an active part, (unless the kernel can recognise when the thread has yielded and the UTS has been entered. (possible I guess) at which time single stepping would be turned off allowing the UTS to run at full speed.

The UTS would hav eto co-operate by using a method of re-entering the thread that allows the kernel to re-start single stepping..

What the other threads are doing in teh meanwhile is unknown.

>  
> *1. keep current ptrace interface, add a command for example  
> PT\_SETDTHREAD to*

freebsd-arch: Re: ptrace and thread

- > *set current thread for debug, and subsequent request for example*
- > *PT\_SETREGS and*
- > *PT\_GETREGS will work on the thread, for single thread process, the*
- > *default current*
- > *thread is always the first thread in the process, this way we needn't*
- > *change legacy debugger*
- > *code.*

yes..

- >
- > *2. introduce a second ptrace syscall, and accept a new parameter tid*
- > *(thread id),*
- > *the PT\_SETREGS and PT\_GETREGS will use the tid to operate on*
- > *corresponding*
- > *thread.*
- >

as mentioned by others this is also what some other systems did.

should this be in -arch or -threads?

- > *For first method, I have a patch there:*
- > *<http://people.freebsd.org/~davidxu/kse/ptrace.diff>*
- > *The patch also includes some bits to support KSE debug, not just for*
- > *pure 1:1 threading.*
- >
- > *David Xu*
- >
- >
- >
- > \_\_\_\_\_
- > *freebsd-arch@freebsd.org mailing list*
- > *<http://lists.freebsd.org/mailman/listinfo/freebsd-arch>*
- > *To unsubscribe, send any mail to "freebsd-arch-unsubscribe@freebsd.org"*
- >

\_\_\_\_\_  
freebsd-arch@freebsd.org mailing list  
<http://lists.freebsd.org/mailman/listinfo/freebsd-arch>  
To unsubscribe, send any mail to "freebsd-arch-unsubscribe@freebsd.org"