

freebsd-current: NULL td passed to propagate\_priority() when using xmms...

## NULL td passed to propagate\_priority() when using xmms...

**Source:** <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2003-10/1603.html>

---

**From:** Sean Chittenden ([sean\\_at\\_chittenden.org](mailto:sean_at_chittenden.org))

**Date:** 10/31/03

Date: Fri, 31 Oct 2003 14:40:23 -0800

To: [current@freebsd.org](mailto:current@freebsd.org)

Howdy. I'm not sure if this is a ULE bug or a KSE bug, or both, but, for those interested (this is using ule 1.67, rebuilding world now), here's my stack. I couldn't figure out where td was being set to NULL. :( Oh! Where is TD\_SET\_LOCK defined? egrep -r didn't turn up anything. -sc

(kgdb) bt

```
#0 doadump () at /usr/src/sys/kern/kern_shutdown.c:240
#1 0xc0530569 in boot (howto=260) at /usr/src/sys/kern/kern_shutdown.c:372
#2 0xc0530948 in panic () at /usr/src/sys/kern/kern_shutdown.c:550
#3 0xc06e6bc6 in trap_fatal (frame=0xd675dc2c, eva=0)
  at /usr/src/sys/i386/i386/trap.c:820
#4 0xc06e6203 in trap (frame=
  {tf_fs = -881065960, tf_es = 16, tf_ds = -881065968, tf_edi = 0, tf_esi = -871763664, tf_ebp =
-696918912, tf_esp = -696918952, tf_ebx = 24, tf_edx = 24, tf_ecx = -871763664, tf_eax = -879614080,
tf_trapno = 12, tf_err = 0, tf_eip = -1068208110, tf_cs = 8, tf_eflags = 66051, tf_esp = -874093984, tf_ss =
0})
  at /usr/src/sys/i386/i386/trap.c:252
#5 0xc06d6a68 in calltrap () at {standard input}:102
#6 0xc05257ac in propagate_priority (td=0x0)
  at /usr/src/sys/kern/kern_mutex.c:152
#7 0xc0525bf9 in _mtx_lock_sleep (m=0xc0796b40, opts=0, file=0x0, line=0)
  at /usr/src/sys/kern/kern_mutex.c:635
#8 0xc051a2c4 in ithread_loop (arg=0xcb7a6b00)
  at /usr/src/sys/kern/kern_intr.c:539
#9 0xc0518f51 in fork_exit (callout=0xc051a100 <ithread_loop>, arg=0x0,
  frame=0x0) at /usr/src/sys/kern/kern_fork.c:796
```

(kgdb) frame 9

```
#9 0xc0518f51 in fork_exit (callout=0xc051a100 <ithread_loop>, arg=0x0,
  frame=0x0) at /usr/src/sys/kern/kern_fork.c:796
```

```
796 callout(arg, frame);
```

(kgdb) list

```
791 * cpu_set_fork_handler intercepts this function call to
792 * have this call a non-return function to stay in kernel mode.
793 * initproc has its own fork handler, but it does return.
```

frebsd-current: NULL td passed to propagate\_priority() when using xmms...

```
794 */
795 KASSERT(callout != NULL, ("NULL callout in fork_exit"));
796 callout(arg, frame);
797
798 /*
799 * Check if a kernel thread misbehaved and returned from its main
800 * function.
(kgdb) frame 8
#8 0xc051a2c4 in ithread_loop (arg=0xcb7a6b00)
  at /usr/src/sys/kern/kern_intr.c:539
539 mtx_lock(&Giant);
(kgdb) list
534 wakeup(ih);
535 mtx_unlock(&ithd->it_lock);
536 goto restart;
537 }
538 if ((ih->ih_flags & IH_MPSAFE) == 0)
539 mtx_lock(&Giant);
540 ih->ih_handler(ih->ih_argument);
541 if ((ih->ih_flags & IH_MPSAFE) == 0)
542 mtx_unlock(&Giant);
543 }
(kgdb) frame 7
#7 0xc0525bf9 in _mtx_lock_sleep (m=0xc0796b40, opts=0, file=0x0, line=0)
  at /usr/src/sys/kern/kern_mutex.c:635
635 propagate_priority(td);
(kgdb) list
630 * Save who we're blocked on.
631 */
632 td->td_blocked = m;
633 td->td_lockname = m->mtx_object.lo_name;
634 TD_SET_LOCK(td);
635 propagate_priority(td);
636
637 if (LOCK_LOG_TEST(&m->mtx_object, opts))
638 CTR3(KTR_LOCK,
639 "_mtx_lock_sleep: p %p blocked on [%p] %s", td, m,
(kgdb) frame 6
#6 0xc05257ac in propagate_priority (td=0x0)
  at /usr/src/sys/kern/kern_mutex.c:152
152 sched_prio(td, pri);
(kgdb) list
147 * XXXKSE this gets a lot more complicated under threads
148 * but try anyhow.
149 */
150 if (TD_ON_RUNQ(td)) {
151 MPASS(td->td_blocked == NULL);
152 sched_prio(td, pri);
153 return;
154 }
155 */
```

NULL td passed to propagate\_priority() when using xmms...

freebsd-current: NULL td passed to propagate\_priority() when using xmms...

156 \* Adjust for any other cases.

```
---
panic: page fault
panic messages:
---
Fatal trap 12: page fault while in kernel mode
fault virtual address   = 0x38
fault code              = supervisor read, page not present
instruction pointer     = 0x8:0xc0547012
stack pointer          = 0x10:0xd6763c6c
frame pointer          = 0x10:0xd6763c80
code segment           = base 0x0, limit 0xfffff, type 0x1b
                       = DPL 0, pres 1, def32 1, gran 1
processor eflags       = interrupt enabled, resume, IOPL = 0
current process       = 33 (irq12: psm0)
trap number           = 12
panic: page fault
syncing disks, buffers remaining...
Fatal trap 12: page fault while in kernel mode
fault virtual address   = 0x38
fault code              = supervisor read, page not present
instruction pointer     = 0x8:0xc0547012
stack pointer          = 0x10:0xd675dc6c
frame pointer          = 0x10:0xd675dc80
code segment           = base 0x0, limit 0xfffff, type 0x1b
                       = DPL 0, pres 1, def32 1, gran 1
processor eflags       = interrupt enabled, resume, IOPL = 0
current process       = 31 (irq5: pcm0)
trap number           = 12
panic: page fault
Uptime: 12h5m43s
Dumping 255 MB
 16 32[CTRL-C to abort] 48 64 80 96 112[CTRL-C to abort] [CTRL-C to abort] [CTRL-C to abort] [CT
---
Reading symbols from /boot/kernel/snd_maestro3.ko...done.
Loaded symbols for /boot/kernel/snd_maestro3.ko
Reading symbols from /boot/kernel/snd_pcm.ko...done.
Loaded symbols for /boot/kernel/snd_pcm.ko
Reading symbols from /usr/obj/usr/src/sys/DELLAPTOP/modules/usr/src/sys/modules/acpi/acpi.ko.debug
Loaded symbols for /usr/obj/usr/src/sys/DELLAPTOP/modules/usr/src/sys/modules/acpi/acpi.ko.debug
#0 doadump () at /usr/src/sys/kern/kern_shutdown.c:240
240          dumping++;
-sc
--
Sean Chittenden
```

---

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