

## Re: sleeping without a mutex on aue(4)

**Source:** <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2005-05/1047.html>

---

**From:** Ian Dowse (*iedowse\_at\_iedowse.com*)

**Date:** 05/31/05

To: yongari@rndsoft.co.kr

Date: Tue, 31 May 2005 09:25:40 +0100

In message <20050531072742.GG4879@rndsoft.co.kr>, Pyun YongHyeon writes:

>Got this panic on recent 6-CURRENT.

...

>#23 0xc059724d in msleep (ident=0xc1362d00, mtx=0x0, priority=76,  
> wmesg=0xc0783676 "usbsyn", tmo=0) at /usr/src/sys/kern/kern\_synch.c:138

>#24 0xc052a0ac in usbd\_transfer (xfer=0xc1362d00)  
> at /usr/src/sys/dev/usb/usbdi.c:344

...

>#33 0xc050b273 in aue\_ioctl (ifp=0xc1419000, command=2149607692, data=0x0)  
> at /usr/src/sys/dev/usb/if\_aue.c:1316

>#34 0xc061fcb5 in in\_ifinit (ifp=0xc1419000, ia=0xc1797d00, sin=0x0, scrub=0)  
>---Type <return> to continue, or q <return> to quit---

> at /usr/src/sys/netinet/in.c:692

>#35 0xc061f276 in in\_control (so=0x0, cmd=1, data=0xc17716c0 "aue0",  
> ifp=0xc1419000, td=0xc1497d80) at /usr/src/sys/netinet/in.c:421

>#36 0xc0610fdd in ifioctl (so=0xc152c7c8, cmd=2151704858, data=0xc17716c0 "aue  
>0",

...

>I guess dropping AUE\_LOCK() before calling usbd\_do\_request() would fix  
>the panic. But is it OK invoking usbd\_do\_request() without a lock  
>held? Should usbd\_xfer\_handle have a pointer to a lock to drop before  
>calling msleep(9)?

The aue driver is Giant-locked like all USB ethernet devices, but some paths to it aren't locking Giant. You could try something like the following – the patch is against an older –CURRENT so might not work directly.

Ian

Index: netinet/in.c

=====  
RCS file: /dump/FreeBSD-CVS/src/sys/netinet/in.c,v

retrieving revision 1.84

diff -u -r1.84 in.c

--- netinet/in.c 20 Mar 2005 14:31:45 -0000 1.84

+++ netinet/in.c 3 Apr 2005 22:48:39 -0000

```

@@ -356,10 +356,15 @@
        return (EINVAL);
        oldaddr = ia->ia_dstaddr;
        ia->ia_dstaddr = *(struct sockaddr_in *)&ifp->ifr_dstaddr;
- if (ifp->if_ioctl && (error = (*ifp->if_ioctl)
- (ifp, SIOCSIFDSTADDR, (caddr_t)ia))) {
- ia->ia_dstaddr = oldaddr;
- return (error);
+ if (ifp->if_ioctl) {
+ IFF_LOCKGIANT(ifp);
+ error = (*ifp->if_ioctl)(ifp, SIOCSIFDSTADDR,
+ (caddr_t)ia);
+ IFF_UNLOCKGIANT(ifp);
+ if (error) {
+ ia->ia_dstaddr = oldaddr;
+ return (error);
+ }
        }
        if (ia->ia_flags & IFA_ROUTE) {
            ia->ia_ifa.ifa_dstaddr = (struct sockaddr *)&oldaddr;
@@ -456,7 +461,10 @@
    default:
        if (ifp == 0 || ifp->if_ioctl == 0)
            return (EOPNOTSUPP);
- return ((*ifp->if_ioctl)(ifp, cmd, data));
+ IFF_LOCKGIANT(ifp);
+ error = (*ifp->if_ioctl)(ifp, cmd, data);
+ IFF_UNLOCKGIANT(ifp);
+ return (error);
    }

    /*
@@ -689,15 +697,19 @@
    * if this is its first address,
    * and to validate the address if necessary.
    */
- if (ifp->if_ioctl &&
- (error = (*ifp->if_ioctl)(ifp, SIOCSIFADDR, (caddr_t)ia))) {
- splx(s);
- /* LIST_REMOVE(ia, ia_hash) is done in in_control */
- ia->ia_addr = oldaddr;
- if (ia->ia_addr.sin_family == AF_INET)
- LIST_INSERT_HEAD(INADDR_HASH(ia->ia_addr.sin_addr.s_addr),
- ia, ia_hash);
- return (error);
+ if (ifp->if_ioctl) {
+ IFF_LOCKGIANT(ifp);
+ error = (*ifp->if_ioctl)(ifp, SIOCSIFADDR, (caddr_t)ia);
+ IFF_UNLOCKGIANT(ifp);
+ if (error) {
+ splx(s);

```

freebsd-current: Re: sleeping without a mutex on aue(4)

```
+ /* LIST_REMOVE(ia, ia_hash) is done in in_control */
+ ia->ia_addr = oldaddr;
+ if (ia->ia_addr.sin_family == AF_INET)
+ LIST_INSERT_HEAD(INADDR_HASH(
+ ia->ia_addr.sin_addr.s_addr), ia, ia_hash);
+ return (error);
+ }
    }
    splx(s);
    if (scrub) {
```

Index: netinet6/in6.c

```
=====
RCS file: /dump/FreeBSD-CVS/src/sys/netinet6/in6.c,v
retrieving revision 1.50
diff -u -r1.50 in6.c
--- netinet6/in6.c 22 Feb 2005 13:04:04 -0000 1.50
+++ netinet6/in6.c 8 Mar 2005 01:19:38 -0000
@@ -1507,10 +1507,14 @@
```

```
    ia->ia_addr = *sin6;

- if (ifacount <= 1 && ifp->if_ioctl &&
- (error = (*ifp->if_ioctl)(ifp, SIOCSIFADDR, (caddr_t)ia))) {
- splx(s);
- return (error);
+ if (ifacount <= 1 && ifp->if_ioctl) {
+ IFF_LOCKGIANT(ifp);
+ error = (*ifp->if_ioctl)(ifp, SIOCSIFADDR, (caddr_t)ia);
+ IFF_UNLOCKGIANT(ifp);
+ if (error) {
+ splx(s);
+ return (error);
+ }
    }
    splx(s);
```

---

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"