

Re: Network Stack Locking

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2004-05/0077.html>

From: Harti Brandt (harti_at_freebsd.org)

Date: 05/24/04

Date: Mon, 24 May 2004 11:08:13 +0200 (MET DST)

To: Robert Watson <rwatson@freebsd.org>

On Thu, 20 May 2004, Robert Watson wrote:

RW>- ATM -- Harti? :-)

Sure. At least netatm, netgraph/atm and the various drivers. At one point I want to get rid of netatm, so I don't want to put effort into netatm (just keep it working and compliling).

netgraph/atm should be clean to the point as netgraph in general is clean (with regards to locking). The drivers (en, [pfh]atm) also are locked, but I'll probably do another iteration when I get my working environment back (I'm currently moving from Berlin to Munich).

RW>- Network device drivers -- some have locking, some have correct locking, RW> some have potential interactions with other pieces of the system (such RW> as the USB stack). Note that for a driver to work correctly with a RW> Giant-free system, it must be safe to invoke ifp->if_start() without RW> holding Giant, and for if_start() to be aware that it cannot RW> acquire Giant without generating a lock order issue. It's OK for RW> if_input() to be called with Giant, although undesirable generally. RW> Some drivers also have locking that is commented out by default due to RW> use of recursive locks, but I'm not sure this is necessarily sufficient RW> problem not to just turn on the locking.

Is there anybody working on the interaction between the network drivers and the module loader (race condition between the interrupt handler and xxx_detach())?

harti

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"