Subject: Re: [BRIDGE] Unaligned access on IA64 when comparing ethernet addresses
Posted by davem on Wed, 18 Apr 2007 20:04:22 GMT

View Forum Message <> Reply to Message

From: Stephen Hemminger <shemminger@linux-foundation.org>
Date: Wed, 18 Apr 2007 07:44:39 -0700

> On Wed, 18 Apr 2007 01:28:04 -0700 (PDT)
> David Miller <davem@davemloft.net> wrote:
>
> > From: Pavel Emelianov <xemul@sw.ru>
> > Date: Wed, 18 Apr 2007 10:43:56 +0400
> >
> > > [snip]
> > >
> > > > --- linux-2.6.orig/net/bridge/br_private.h 2007-04-17
> > > > 13:26:48.000000000 -0700 +++ linux-2.6/net/bridge/br_private.h
> > > > 2007-04-17 13:30:29.000000000 -0700 @@ -36,7 +36,7 @@
> > > > {
> > > >   unsigned char prio[2];
> > > >   unsigned char addr[6];
> > > > -};
> > > > +} __attribute__((aligned(8)));
> > >
> > > Why "8"? Mustn't it be "16"? Address is to be 2-bytes aligned...
> >
> > Actually it could be made "2", the aligned() attribute is
> > in bytes, not bits.
>
> It could be 2 but 8 might allow a compiler on a 64 bit platform
> to be smarter in comparisons and assignments.

Absolutely.

Although I don't think gcc does anything fancy since we don't
use memcmp(). It's a tradeoff, we'd like to use unsigned long
comparisons when both objects are aligned correctly but we also
don't want it to use any more than one potentially mispredicted
branch.

We could add some alignment tests to the ethernet address
comparison code, but it's probably more trouble than it's
worth.