Subject: Re: [BRIDGE] Unaligned access on IA64 when comparing ethernet addresses
Posted by davem on Thu, 19 Apr 2007 20:01:01 GMT
View Forum Message <> Reply to Message

From: Eric Dumazet <dada1@cosmosbay.com>
Date: Thu, 19 Apr 2007 16:14:23 +0200

> On Wed, 18 Apr 2007 13:04:22 -0700 (PDT)
> David Miller <davem@davemloft.net> wrote:
>
> >
> > 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.
>
> Again, memcmp() *cannot* be optimized, because its semantic is to compare bytes.
>
> memcpy() can take into account alignement if known at compile time, not memcmp()
>
> http://lists.openwall.net/netdev/2007/03/13/31

I was prehaps thinking about strlen() where I know several
implementations work a word at a time even though it is
a byte-based operation:

```
--------------------
#define LO_MAGIC 0x01010101
#define HI_MAGIC 0x80808080
 ...
  sethi %hi(HI_MAGIC), %o4
 ...
  or %o4, %lo(HI_MAGIC), %o3
 ...
  sethi %hi(LO_MAGIC), %o4
 ...
  or %o4, %lo(LO_MAGIC), %o2
 ...
8:
 ld [%o0], %o5
2:
 sub %o5, %o2, %o4
 andcc %o4, %o3, %g0
 be,pt %icc, 8b
  add %o0, 4, %o0
--------------------
```

I figured some similar trick could be done with strcmp() and memcmp().