

Re: [current tinderbox] failure on ...all...

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2005-06/0542.html>

From: Ruslan Ermilov (ru_at_freebsd.org)

Date: 06/10/05

Date: Fri, 10 Jun 2005 12:46:15 +0300

To: Dag-Erling Sm?rgrav <des@des.no>, Joseph Koshy <joseph.koshy@gmail.com>, current@freebsd.org

On Fri, Jun 10, 2005 at 11:16:27AM +0200, Stefan Farfeleder wrote:

> On Fri, Jun 10, 2005 at 11:06:16AM +0200, Dag-Erling Sm?rgrav wrote:

> > Joseph Koshy <joseph.koshy@gmail.com> writes:

> > > Dag-Erling Sm?rgrav <des@des.no> writes:

> > > > It also seems strange to me that you on the one hand introduce a

> > > > new struct to separate MD and MI interfaces, and on the other hand

> > > > continue to assume that they are assignment-compatible.

> > > I'd be very surprised if two C structures with identical definitions

> > > were not assignment compatible.

> >

> > I wouldn't be surprised if the standard says they aren't.

> > Unfortunately, my copy is at home.

>

> Do you mean the following?

>

> struct t1 { int a; } x;

> struct t2 { int a; } y = { 42 };

> x = y;

>

> The types `struct t1' and `struct t2' are not compatible and thus not

> assignable. See 6.2.7 and 6.5.16.1.

>

If you're to byte-copy say t1 to t2, is it guaranteed to work? That is, do both types are guaranteed to have the same size and alignment of their structure members? I'm pretty sure this is guaranteed, as lot of code assumes this, for example, the sockaddr* structures.

Cheers,

--

Ruslan Ermilov
ru@FreeBSD.org
FreeBSD committer

• application/pgp-signature attachment: [stored](#)