

Re: Global (non _KERNEL) place for sockaddr_union?

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/hackers/2004-09/0275.html>

From: Brooks Davis (brooks_at_one-eyed-alien.net)

Date: 09/20/04

Date: Sun, 19 Sep 2004 21:24:17 -0700

To: Max Laier <max@love2party.net>

On Mon, Sep 20, 2004 at 05:39:38AM +0200, Max Laier wrote:

> On Monday 20 September 2004 04:28, Brooks Davis wrote:

>> On Mon, Sep 20, 2004 at 02:50:40AM +0200, Max Laier wrote:

>>> Hi,

>>>

>>> <http://www.freebsd.org/cgi/query-pr.cgi?pr=kern/71836> is the symptom. Now

>>> I am looking for a clean solution to it. What is needed is an include

>>> file that defines union sockaddr_union in a way that is useable from

>>> kernel and userland. Historically it seems that this union first appeared

>>> in context of ipsec within the kernel. pf has adopted it, but uses it in

>>> the userland as well. I am sure that it can be usefull in a lot of places

>>> that have to deal with/store different address formats.

>>>

>>> My question now is, what would be a good place to define this? Are there

>>> any fromal standarts that might define it already? (Couldn't find

>>> anything) Is there anything else that I must consider?

>>>

>>> At some point I though netinet/in.h might be a good place, but that'd

>>> require inclusion of sys/socket.h, which certainly is not a good

>>> solution.

>>>

>>> Opinions? Ideas?

>>>

>>>> #include <sys/socket.h>

>>>> #include <netinet/in.h>

>>>>

>>>> union sockaddr_union {

>>>> struct sockaddr sa;

>>>> struct sockaddr_in sin;

>>>> struct sockaddr_in6 sin6;

>>>> struct sockaddr_storage __su_pad; /* maybe not a bad idea */

>>>> };

>>>

>> I don't see an elegant solution. Stuffing it off in its own file may

>> be the best thing if you're going to use it. Overall, I'd say it's bad

freebsd-hackers: Re: Global (non _KERNEL) place for sockaddr_union?

> > *idea that PF be better off without. It appears to save a few casts,*
> > *but nothing worth the pain of generalizing the declaration.*
>
> *First of all, the padding is bogus as sin6 is big enough. Especially since one*
> *point here is to save space. I was a bit confused there, sorry. Especially*
> *since this is an important point: In pf this union is used to – for example –*
> *store address information in tables. It allows to store IPv4 and IPv6*
> *addresses in the same table without creating overhead in the memory footprint*
> *or having to deal with different objects for every address type. The fewer*
> *casts are just an additional benefit.*
>
> *Maybe you are right and a new header is the easiest way out. Moving this out*
> *of under _KERNEL would require all inclusions of netipsec/keydb.h to include*
> *sys/socket.h andnetinet/in.h. As I was saying, I don't have a good idea*
> *either. The only thing that came to my mind just now is to add a protecting*
> *define and #ifdef around the two places that define it. But I have no idea*
> *how clean (in terms of style) such a solution is.*

If it's primarily for well defined storage, why not remove the sa and put it innetinet/in.h? If you're just looking for the family and length, it doesn't matter if you access it as .sa, .sin, or .sin6. You might have to name it something else which could be problematic for portability, but it might be worth the pain if you could push it back to OpenBSD.

-- Brooks

-- Brooks

--

Any statement of the form "X is the one, true Y" is FALSE.
PGP fingerprint 655D 519C 26A7 82E7 2529 9BF0 5D8E 8BE9 F238 1AD4

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