
Subject: Re: [PATCH 4/6] BC: user interface (syscalls)

Posted by [dev](#) on Fri, 25 Aug 2006 10:54:18 GMT

[View Forum Message](#) <> [Reply to Message](#)

Alexey Dobriyan wrote:

> On Thu, Aug 24, 2006 at 12:04:16PM +0100, Alan Cox wrote:

>

>>Ar Mer, 2006-08-23 am 21:35 -0700, ysgrifennodd Andrew Morton:

>>

>>>>Its a uid_t because of setluid() and twenty odd years of existing unix
>>>>practice.

>>>>

>>>

>>>I don't understand. This number is an identifier for an accounting
>>>container, which was somehow dreamed up by userspace.

>>

>>Which happens to be a uid_t. It could easily be anyother_t of itself and
>>you can create a container_id_t or whatever. It is just a number.

>>

>>The ancient Unix implementations of this kind of resource management and
>>security are built around setluid() which sets a uid value that cannot
>>be changed again and is normally used for security purposes. That
>>happened to be a uid_t and in simple setups at login uid = luid = euid
>>would be the norm.

>>

>>Thus the Linux one happens to be a uid_t. It could be something else but
>>for the "container per user" model whatever a container is must be able
>>to hold all possible uid_t values. So we can certainly do something like

>>

>>typedef uid_t container_id_t;

>

>

> What about cid_t? Google mentions cid_t was used in HP-UX specific IPC (only if
> _INCLUDE_HPUX_SOURCE is defined).

bcid_t?

Thanks,

Kirill
