Subject: Re: [PATCH] An attempt to have an unlimitedly extendable sys_clone
Posted by Dave Hansen on Tue, 15 Jan 2008 17:54:20 GMT
View Forum Message <> Reply to Message

On Tue, 2008-01-15 at 15:50 +0300, Pavel Emelyanov wrote:

```
> +static struct long_clone_arg *get_long_clone_arg(int __user
> *child_tidptr)
> +{
> +	int size;
> +	struct long_clone_arg *carg;
> +
> +	if (get_user(size, child_tidptr))
> +		return ERR_PTR(-EFAULT);
> +
> +	if (size > sizeof(struct long_clone_arg))
> +		return ERR_PTR(-EINVAL);
```

This little bit means that any newer app (with a large
long_clone_arg->size) trying to run on an older kernel (with a smaller
struct) would simply fail to run clone(). Perhaps it shouldn't be _so_
generic as to allow anything in the struct and should stick to bits.
That way, we can actually go look to see whether there are any _unknown_
bits set just like we do now with clone flags.

The more I think about this, the more nervous I get about it. It is
really neat, but has a bit of the stink of ioctl()s on it. I'd
personally rather see a new system call.

But, this seems like a good Linus question. Want to keep us on cc, but
run it by him (and the rest of LKML)?

-- Dave

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers