

---

Subject: Re: [patch -mm 10/17] nsproxy: add unshare\_ns and bind\_ns syscalls  
Posted by [ebiederm](#) on Sat, 09 Dec 2006 07:40:03 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Herbert Poetzl <herbert@13thfloor.at> writes:

> On Fri, Dec 08, 2006 at 12:26:49PM -0700, Eric W. Biederman wrote:  
>> clg@fr.ibm.com writes:  
>>  
>> > From: Cedric Le Goater <clg@fr.ibm.com>  
>> >  
>> > The following patch defines 2 new syscalls specific to nsproxy and  
>> > namespaces :  
>> >  
>> > \* unshare\_ns :  
>> >  
>> > enables a process to unshare one or more namespaces. this  
>> > duplicates the unshare syscall for the moment but we  
>> > expect to diverge when the number of namespaces increases  
>>  
>> Are we out of clone flags yet? If not this is premature.  
>  
> no, but a different nevertheless related question:  
> does anybody, except for 'us' use the unshare() syscall?

The pam\_namespace module if I have looked at things properly. I believe that is what it was added to support.

> because if not, then why not simply extend that one  
> to 64bit and be done, we probably won't need a clone64()  
> but if we find we do (at some point) adding that with  
> the new flags would be trivial ...  
>  
> OTOH, we could also just add an unshare64() too  
>  
> anyway, we \_will\_ run out of flags in the near future

Agreed. Please let's cross that bridge when we come to it.

>> I'm also worried about the security implications of switching  
>> namespaces on a process.  
>> That is something that needs to be looked at very closely.  
>  
> Linux-VServer currently uses a capability to prevent  
> changing between namespaces (a very generic one) but  
> it probably makes sense to add something like that  
> in general ... btw, did I mention that the capability  
> flags are running out too?

I think they have run out. Not that sys\_capability needs a revision but it appears the format of the data does which is likely just as bad.

>> These two changes certainly don't belong in a single patch,  
>> and they certainly use a bit more explanation.  
>> syscalls are not something to add lightly.  
>  
> well, and they will take ages to get into mainline  
> for all archs, or has that changed since we reserved  
> sys\_vserver()?

I think it is likely a little better. I'm not certain what your definition of ages is.

>> Because they must be supported forever.  
>  
> I'm not sure about that, most archs 'reuse' syscalls  
> when there is no user left ...

I haven't seen that on i386. Except for experimental syscalls I have a hard time believing we have any syscalls that have had all of their users disappear.

Reusing syscall numbers is in a lot of ways completely irresponsible once you start supporting a binary interface. Even if you remove the syscall because there are no users or it makes absolutely no sense any more.

Eric

---

Containers mailing list  
Containers@lists.osdl.org  
<https://lists.osdl.org/mailman/listinfo/containers>

---