Subject: Re: [PATCH 11/11] protect architectures where THREAD_SIZE >= PAGE_SIZE against fork bombs
Posted by David Rientjes on Tue, 26 Jun 2012 08:45:30 GMT

On Tue, 26 Jun 2012, Glauber Costa wrote:

> > > diff --git a/include/linux/thread_info.h b/include/linux/thread_info.h
> > > index ccc1899..914ec07 100644
> > > --- a/include/linux/thread_info.h
> > > +++ b/include/linux/thread_info.h
> > > @@ -61,6 +61,12 @@ extern long do_no_restart_syscall(struct restart_block
> > > *parm);
> > >   # define THREADINFO_GFP (GFP_KERNEL | __GFP_NOTRACK)
> > >   #endif
> > >
> > > +#ifdef CONFIG_CGROUP_MEM_RES_CTLR_KMEM
> > > +# define THREADINFO_GFP_ACCOUNTED (THREADINFO_GFP | __GFP_KMEMCG)
> > > +#else
> > > +# define THREADINFO_GFP_ACCOUNTED (THREADINFO_GFP)
> > > +#endif
> > > +
> >
> > This type of requirement is going to become nasty very quickly if nobody
> > can use __GFP_KMEMCG without testing for
CONFIG_CGROUP_MEM_RES_CTLR_KMEM.
> > Perhaps define __GFP_KMEMCG to be 0x0 if it's not enabled, similar to how
> > kmemcheck does?
> >
> That is what I've done in my first version of this patch. At that time,
> Christoph wanted it to be this way so we would make sure it would never be
> used with #CONFIG_CGROUP_MEM_RES_CTLR_KMEM defined. A value of zero will
> generate no errors. Undefined value will.
>
> Now, if you ask me, I personally prefer following what kmemcheck does here...
>

Right, because I'm sure that __GFP_KMEMCG will be used in additional
places outside of this patchset and it will be a shame if we have to
always add #ifdef's. I see no reason why we would care if __GFP_KMEMCG
was used when CONFIG_CGROUP_MEM_RES_CTLR_KMEM=n with the semantics that it
as in this patchset. It's much cleaner by making it 0x0 when disabled.