Subject: Re: [PATCH v5 13/14] protect architectures where THREAD_SIZE >= PAGE_SIZE against fork bombs
Posted by Glauber Costa on Thu, 18 Oct 2012 09:37:39 GMT
View Forum Message <> Reply to Message

On 10/18/2012 02:12 AM, Andrew Morton wrote:
> On Tue, 16 Oct 2012 14:16:50 +0400
> Glauber Costa <glommer@parallels.com> wrote:
>
>> @@ -146,7 +146,7 @@ void __weak arch_release_thread_info(struct thread_info *ti)
>> static struct thread_info *alloc_thread_info_node(struct task_struct *tsk,
>> int node)
>> {
>> - struct page *page = alloc_pages_node(node, THREADINFO_GFP,
>> + struct page *page = alloc_pages_node(node, THREADINFO_GFP_ACCOUNTED,
>> THREAD_SIZE_ORDER);
>>
> yay, we actually used all this code for something ;)
>
Happy to be of use, sir!

> I don't think we really saw a comprehensive list of what else the kmem
> controller will be used for, but I believe that all other envisaged
> applications will require slab accounting, yes?
>
>
> So it appears that all we have at present is a
> yet-another-fork-bomb-preventer, but one which requires that the
> culprit be in a container? That's reasonable, given your
> hosted-environment scenario. It's unclear (to me) that we should merge
> all this code for only this feature. Again, it would be good to have a
> clear listing of and plan for other applications of this code.
>

I agree. This doesn't buy me much without slab accounting. But
reiterating what I've just said in another e-mail, slab accounting is
not really in plan stage, but had also been through extensive development.

As a matter of fact, it used to be only "slab accounting" in the
beginning, without this. I've split it more recently because I believe
it would allow people to do a more focused review, leading to better code.