Subject: Re: [PATCH v5 13/14] protect architectures where THREAD_SIZE >= PAGE_SIZE against fork bombs
Posted by akpm on Wed, 17 Oct 2012 22:12:45 GMT
View Forum Message <> Reply to Message

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 ;)

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.