

---

Subject: Re: [PATCH] Fix some kallsyms\_lookup() vs rmmod races  
Posted by [Alexey Dobriyan](#) on Thu, 15 Mar 2007 18:26:53 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Thu, Mar 15, 2007 at 04:53:59PM +0000, Paulo Marques wrote:

> Alexey Dobriyan wrote:  
> >[cc'ing folks whose proc files are affected]  
> >  
> >kallsyms\_lookup() can call module\_address\_lookup() which iterates over  
> >modules list without module\_mutex taken. Comment at the top of  
> >module\_address\_lookup() says it's for oops resolution so races are  
> >irrelevant, but in some cases it's reachable from regular code:  
>  
> So maybe we should just add a new parameter to "kallsyms\_lookup" to  
> inform it if it is safe to take a mutex or not.

You have to drag "mod->name" out of kallsyms\_lookup(), so if you drop module\_mutex in it, you still have a bug.

We can agree on kallsyms\_lookup() or whatever other low-level function to copy everything into caller-supplied structure and not spreading module\_mutex. module's name is 64 minus a little, so stack usage should be fine.

> Spreading module\_mutex everywhere doesn't seem like the right interface  
> for several reasons:  
>  
> - new users of "kallsyms\_lookup" might not be aware that they should  
> take module\_mutex if it is safe

Well, yes.

> - many times we will be taking module\_mutex even when we are fetching  
> a kernel symbol that shouldn't require the mutex at all  
>  
> - it just creates new dependencies (hint: this patch shouldn't even  
> compile with current git since module\_mutex is not declared in module.h,

Yeah, I remembered about it only in subway :).

This patch is techically dependent on <http://lkml.org/lkml/2007/3/14/128> aka [PATCH v2] Race between cat /proc/kallsyms and rmmod

> not to mention compile when CONFIG\_MODULES not set)

OK, I'll fix it.

> IMHO we should not expose module\_mutex outside of module.c. That is just

> wrong from an encapsulation point of view.

---