
Subject: Re: Racy /proc creations interfaces

Posted by [adobriyan](#) on Thu, 28 Dec 2006 08:15:37 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Wed, Dec 27, 2006 at 01:56:24PM +0000, Al Viro wrote:

> On Wed, Dec 27, 2006 at 04:42:23PM +0300, Alexey Dobriyan wrote:

> >

> > struct proc_entry_raw foo_pe_raw = {

> > .owner = THIS_MODULE,

> > .name = "foo",

> > .mode = 0644,

> > .read_proc = foo_read_proc,

> > .data = foo_data,

> > .parent = foo_parent,

> > };

> >

> > pde = create_proc_entry(&foo_pe_raw);

> > if (!pde)

> > return -ENOMEM;

> >

> > where "struct proc_entry_raw" is cut down version of "struct proc_dir_entry"

>

> Ewwwwwwwwwwwwwwwwww

>

> Please, please no. Especially not .parent. If anything, let's add a

> helper saying "it's all set up now". And turn create_proc_entry()

> into a macro that would pass THIS_MODULE to underlying function and

> call that helper, so that simple cases wouldn't have to bother at all.

People are setting ->data after create_proc_entry():

drivers/zorro/proc.c:

```
110 static int __init zorro_proc_attach_device(u_int slot)
```

```
111 {
```

```
112 struct proc_dir_entry *entry;
```

```
113 char name[4];
```

```
114
```

```
115 sprintf(name, "%02x", slot);
```

```
116 entry = create_proc_entry(name, 0, proc_bus_zorro_dir);
```

```
117 if (!entry)
```

```
118 return -ENOMEM;
```

```
119 entry->proc_fops = &proc_bus_zorro_operations;
```

```
120 entry->data = &zorro_autocon[slot];
```

```
121 entry->size = sizeof(struct zorro_dev);
```

If create_proc_entry is a macro doing what you suggest (am I right?)

```
#define create_proc_entry(name, mode, parent)
```

```
({  
    struct proc_dir_entry *pde;  
  
    pde = __create_proc_entry(name, mode, parent, THIS_MODULE);  
    if (pde)  
        mark_proc_entry_ready(pde);  
    pde;  
})
```

there is still a problem because we want it to be equivalent to

```
pde = create_proc_entry(...);  
if (!pde)  
    return -ENOMEM;  
pde->proc_fops = ...;  
pde->data = ...;  
mark_proc_entry_ready(pde);
```
