

Re: [BUG 2.6.20-rc2-mm1] init segfaults when CONFIG\_PROFILE\_LIKELY=y

## Re: [BUG 2.6.20-rc2-mm1] init segfaults when CONFIG\_PROFILE\_LIKELY=y

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-12/msg08184.html>

---

- *From:* Randy Dunlap <[rdunlap@xxxxxxxxxxxxx](mailto:rdunlap@xxxxxxxxxxxxx)>
  - *Date:* Sun, 31 Dec 2006 12:43:58 -0800
- 

On Sun, 31 Dec 2006 11:45:09 -0800 Daniel Walker wrote:

On Sun, 2006-12-31 at 23:04 +0800, Fengguang Wu wrote:

Hi,

The following messages keeps popping up when CONFIG\_PROFILE\_LIKELY=y:

```
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
init[1]: segfault at ffffffff8118c110 rip ffffffff8118c110 rsp
00007fff9a9d14d8 error 15
```

Does this seem like an appropriate solution? This just reconstitutes Ingo's patch by removing the unlikely calls that got added recently.

How does this fix the problem? (if it does)  
What is the real cause of the problem?

Re: [BUG 2.6.20-rc2-mm1] init segfaults when CONFIG\_PROFILE\_LIKELY=y

Maybe a comment into vsyscall.c that says to stay away from all macro's and possible debug code that could be added might be helpful ?

Why?

Signed-Off-By: Daniel Walker <dwalker@xxxxxxxxxxx>

---  
arch/x86\_64/kernel/vsyscall.c | 4 ++--  
1 files changed, 2 insertions(+), 2 deletions(-)

Index: linux-2.6.19/arch/x86\_64/kernel/vsyscall.c

```
=====
--- linux-2.6.19.orig/arch/x86_64/kernel/vsyscall.c
+++ linux-2.6.19/arch/x86_64/kernel/vsyscall.c
@@ -111,7 +111,7 @@ static __always_inline void do_vgettimeo
seq = read_seqbegin(&__vsyscall_gtod_data.lock);

vread = __vsyscall_gtod_data.clock.vread;
- if (unlikely(!__vsyscall_gtod_data.sysctl_enabled || !vread)) {
+ if (!__vsyscall_gtod_data.sysctl_enabled || !vread) {
gettimeofday(tv,0);
return;
}
@@ -151,7 +151,7 @@ int __vsyscall(0) vgettimeofday(struct t
* unlikely */
time_t __vsyscall(1) vtime(time_t *t)
{
- if (unlikely(!__vsyscall_gtod_data.sysctl_enabled))
+ if (!__vsyscall_gtod_data.sysctl_enabled)
return time_syscall(t);
else if (t)
*t = __vsyscall_gtod_data.wall_time_tv.tv_sec;
```

---  
~Randy

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxxx  
More majordomo info at <http://vger.kernel.org/majordomo-info.html>  
Please read the FAQ at <http://www.tux.org/lkml/>