

---

Subject: [PATCH] /proc/\*/oom\_score oops re badness  
Posted by [Alexey Dobriyan](#) on Thu, 15 Mar 2007 09:01:51 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Eternal quest to make

```
while true; do cat /proc/fs/xfs/stat >/dev/null 2>/dev/null; done  
while true; do find /proc -type f 2>/dev/null | xargs cat >/dev/null 2>/dev/null; done  
while true; do modprobe xfs; rmmod xfs; done
```

work reliably continues and now kernel[1] oopses in the following way:

```
BUG: unable to handle ... at virtual address 6b6b6b6b  
EIP is at badness  
process: cat  
proc_oom_score  
proc_info_read  
sys_fstat64  
vfs_read  
proc_info_read  
sys_read
```

Failing code is prefetch hidden in list\_for\_each\_entry() in badness().  
badness() is reachable from two points. One is proc\_oom\_score, another  
is out\_of\_memory() => select\_bad\_process() => badness().

Second path grabs tasklist\_lock, while first doesn't.

[1] 2.6.21-rc3-something + patches for all proc races I've already  
posted + patch for proxying proc directories similar to proxying  
proc regular files.

Signed-off-by: Alexey Dobriyan <[adobriyan@sw.ru](mailto:adobriyan@sw.ru)>

---

```
fs/proc/base.c | 2 ++  
1 file changed, 2 insertions(+)
```

```
--- a/fs/proc/base.c  
+++ b/fs/proc/base.c  
@@ -310,7 +310,9 @@ static int proc_oom_score(struct task_st  
    struct timespec uptime;  
  
    do_posix_clock_monotonic_gettime(&uptime);  
+ read_lock(&tasklist_lock);  
    points = badness(task, uptime.tv_sec);  
+ read_unlock(&tasklist_lock);  
    return sprintf(buffer, "%lu\n", points);
```

}

---