

Race Condition over sys_tz

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-09/msg08963.html>

- *From:* "Dong Feng" <middle.fengdong@xxxxxxxxx>
 - *Date:* Sat, 30 Sep 2006 08:53:24 +0800
-

The operations on sys_tz, so far known to me in sys_settimeofday and sys_gettimeofday, is neither atomic nor protected by any lock. I suspect it probably causes unpredictable behavior when multiple processes try to set the system time zone simultaneously.

Following is the code fragment extracted from do_sys_settimeofday(). The function is invoked by sys_settimeofday() without locking. At least two non-atomic operations:

1. struct copy between *tz and sys_tz.
2. The test-and-operate over firsttime.

```
if (tz) {
/* SMP safe, global irq locking makes it work. */
sys_tz = *tz;
if (firsttime) {
firsttime = 0;
if (!tv)
warp_clock();
}
}
-
```

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>