

## 2.6: Load average calculation?

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-03/msg09541.html>

---

- *From:* Russell King <[rmk+lkml@xxxxxxxxxxxxxxxxxxx](mailto:rmk+lkml@xxxxxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 28 Mar 2006 11:56:12 +0100
- 

Hi,

2.6.11 based FC3 kernel.

One of the servers being used to download FC5 has a rather high load average, which is hardly surprising. What is, is the results from top and the fact that the machine is still *\_very\_* responsive via ssh.

```
top - 11:05:22 up 206 days, 22:31, 49 users, load average: 145.78, 140.34, 133
Tasks: 1221 total, 10 running, 1209 sleeping, 2 stopped, 0 zombie
Cpu(s): 1.4% us, 0.6% sy, 0.0% ni, 93.0% id, 4.4% wa, 0.5% hi, 0.0% si
Mem: 2075852k total, 2025220k used, 50632k free, 10152k buffers
Swap: 2249060k total, 576k used, 2248484k free, 1032408k cached
```

Note the high load average and the mostly idle (not io wait) CPU %age.

The load average seems to be coming from the apache / vsftpd:

```
PID USER STAT COMMAND WCHAN
818 apache D httpd sync_buffer
6517 apache D httpd sync_buffer
6527 apache D httpd sync_page
6575 apache D httpd sync_page
1774 ftp D vsftpd sync_page
... about 128 more vsftpds in the same state ...
9335 ftp D vsftpd sync_page
```

and if we look closer, from sysrq-t (note that the dump is far larger than the system log buffer, so I will only give a couple of examples):

```
vsftpd D D72C6DF4 2452 12309 12306 (NOTLB)
d72c6e20 00000082 c013d4c9 d72c6df4 d72c6df4 f7c01d48 c028a0cc f7c01d48
c028a13a e5bd0db0 c013d4c9 d72c6df4 d72c6df4 00000202 00000246 00000000
7adc5940 004ecb04 e5bd0f18 d72c6e70 d72c6e78 c200da20 d72c6e28 c0364c53
Call Trace:
[<c013d4c9>] autoremove_wake_function+0x0/0x37
[<c028a0cc>] __generic_unplug_device+0x16/0x31
[<c028a13a>] generic_unplug_device+0x53/0x158
[<c013d4c9>] autoremove_wake_function+0x0/0x37
```

## 2.6: Load average calculation?

```
[<c0364c53>] io_schedule+0xe/0x16
[<c014afec>] sync_page+0x36/0x42
[<c0364f57>] __wait_on_bit_lock+0x3e/0x5e
[<c014afb6>] sync_page+0x0/0x42
[<c014b7b9>] __lock_page+0x90/0x98
[<c013d500>] wake_bit_function+0x0/0x3c
[<c013d500>] wake_bit_function+0x0/0x3c
[<c01a5a55>] mpage_readpage+0x39/0x3f
[<c014c4c0>] do_generic_mapping_read+0x3ae/0x63d
[<c014cbcd>] generic_file_sendfile+0x5e/0x70
[<c014cb2d>] file_send_actor+0x0/0x62
[<c01759fd>] do_sendfile+0x1d3/0x28e
[<c014cb2d>] file_send_actor+0x0/0x62
[<c0175b6f>] sys_sendfile+0xb7/0xc2
[<c0103903>] syscall_call+0x7/0xb
```

vsftpd D 00000100 2452 12353 12351 (NOTLB)

```
c98d9e20 00000082 cdfc7780 00000100 ea4bce80 f7c01d48 c028a0cc f7c01d48
c028a13a ea4bce80 00000000 c032368a c98d9df4 00000202 00000246 00000000
2b782400 004ecb05 f4d7cc98 c98d9e70 c98d9e78 c2016240 c98d9e28 c0364c53
```

Call Trace:

```
[<c028a0cc>] __generic_unplug_device+0x16/0x31
[<c028a13a>] generic_unplug_device+0x53/0x158
[<c032368a>] do_tcp_sendpages+0x3ce/0xa25
[<c0364c53>] io_schedule+0xe/0x16
[<c014afec>] sync_page+0x36/0x42
[<c0364f57>] __wait_on_bit_lock+0x3e/0x5e
[<c014afb6>] sync_page+0x0/0x42
[<c014b7b9>] __lock_page+0x90/0x98
[<c013d500>] wake_bit_function+0x0/0x3c
[<c013d500>] wake_bit_function+0x0/0x3c
[<c014c4b4>] do_generic_mapping_read+0x3a2/0x63d
[<c014cbcd>] generic_file_sendfile+0x5e/0x70
[<c014cb2d>] file_send_actor+0x0/0x62
[<c01759fd>] do_sendfile+0x1d3/0x28e
[<c014cb2d>] file_send_actor+0x0/0x62
[<c0175b6f>] sys_sendfile+0xb7/0xc2
[<c0103903>] syscall_call+0x7/0xb
```

The disk subsystem is coping very well with this load. However, the network interface through which all the ftp and http traffic is flowing is running at around 92mbps (timed over 10 seconds), and is therefore probably close to saturation. (Note that this is the same network interface through which ssh is connected, which remains responsive.)

So far so good.

However, programs such as MTAs make decisions about delivery based on the load average, so a high induced (but apparently fictitious) load average denies service to other parts of the system.

## 2.6: Load average calculation?

So, the question becomes – should a lot of network activity contribute to the system load average, thereby denying other services from performing their usual business.

--

Russell King

Linux kernel 2.6 ARM Linux – <http://www.arm.linux.org.uk/>

maintainer of: 2.6 Serial core

–

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

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

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