Tuning Tier 2 Systems for TCP/IP Performance on High Latency Networks

WLCG Collaboration Workshop (Tier0/Tier1/Tier2)
22-26 January 2007

Mark Bowden, Wenji Wu, Matt Crawford
(bowden@fnal.gov, wenji@fnal.gov, crawdad@fnal.gov)

Fermilab’s Wide Area Networking group performed a number of experiments and investigations to understand how to improve data transfers in high latency networks

This resulted in concrete recommendations that allowed sites to improve transfer rates while increasing system stability.

I present only some of the conclusions

Please see the detailed findings at http://indico.cern.ch/getFile.py/access?contribId=29&sessionId=6&resId=0&materialId=slides&confId=5490

Conclusions

with anecdotal comments from Google

- 32 bit Linux has insufficient kernel space for large networking applications

  -default tcp_mem chosen badly for large-memory (fixed in 2.6.19)

  - switch to a 64 bit machine if possible

  - limit maximum TCP buffer sizes

  - limit number of connections per machine

  - consider separate machines for local and remote transfers

  - try “4G/4G” kernel (not mainline, possible incompatibilities, slower)

> “Oh, the answer is very simple: it's not going to happen. EVER. You need more than 32 bits of address space to handle that kind of memory. This is not something I'm going to discuss further...This is not negotiable.”
>
> Linus Torvalds - 1999 (on running 32 bit Linux with more than 2GB of memory)

Conclusions

- Networking code in Linux 2.6 kernels is much poorer than 2.4

“Using the 2.6 kernel [vs 2.4] on embedded systems implicates the following disadvantages:
...context switches up to **96%** slower, local communication latencies up to **80%** slower, file system latencies up to **76%** slower, local communication bandwidth **less than 50%** in some cases...”

http://www.denx.de/wiki/Know/Linux24vs26

“...found through his studies that kernel 2.4...outperformed kernel 2.6 in every test and under every configuration in terms of throughput and thus concluded that kernel 2.6 still has room for improvement.”

http://os.newsforge.com/article.pl?sid=05/07/22/1054216&tid=2&tid=18

“Is there any chance Linus will freeze 2.6 and make the current development tree 2.7? It seems like ever since around 2.6.8 things have been getting progressively worse (page allocation failures, ...”

http://www.gatago.com/linux/kernel/15485908.html

“FreeBSD 4.9: Drops no packets at 900K pps
Linux 2.4.24: Starts dropping packets at 350K pps
Linux 2.6.12: Starts dropping packets at 100K pps”

http://www.gatago.com/linux/kernel/14693564.html

Conclusions

- Linux network RX has low priority...packet drops are “expected”
  - limit TX and other activity during RX
  - modify driver to improve RX priority
  - wait for massive rewrite

“... if the RX path gets very busy packets will end up being dropped ... by virtue of DMA rings being filled up ... and that is an acceptable compromise.”

http://www.spinics.net/lists/netdev/msg10601.html

“The issue is that RX is absolute, as you cannot "decide" to delay or selectively drop since you don't know what’s coming. Better to have some latency than dropped packets. But if you don't dedicate to RX, then you have an unknown amount of cpu resources doing "other stuff". The capacity issues always first manifest themselves as RX overruns, and they always happen a lot sooner on MP machines than UP machines. The LINUX camp made the mistake of not making RX important enough, and now their 2.6 kernels drop packets all over the ranch. But audio is nice and smooth...”

http://leaf.dragonflybsd.org/mailarchive/users/2005-12/msg00007.html

Conclusions

- Linux memory management is poor, especially for network applications
  - limit swapping
  - reduce(!) RAM (increases available low mem, reduces unnecessary disk caching)

> “...there's some latency involved... if the CPU is stuck in an interrupt handler refilling a huge network RX ring, then waking kswapd won't do anything and you will run out of memory.”
>
> http://oss.sgi.com/archives/xfs/2004-12/msg00008.html

> “The bug density in Linux and its programs remains an embarrassment ...
> As bug nests go, Linux memory/swap management probably remains its greatest core problem...”
>
> http://lwn.net/Articles/140257/

The Fermilab wide area networking group is committed to direct support of CMS, but will also work with and assist any LHC Tier2 site as time permits.

Please contact us with any interesting network problems…at worst we can tell you that your problem is novel, at best we may be able to show that it's been solved. (If you are a network researcher, reverse "worst" and "best.")

Please see the detailed presentation given at at WLHC Computing Grid Tier-2 Workshop in Asia
http://indico.cern.ch/getFile.py/access?contribId=29&sessionId=6&resId=0&materialId=slides&confId=5490

Matt Crawford (crawdad@fnal.gov)
Wenji Wu (wenji@fnal.gov)
Mark Bowden (bowden@fnal.gov)
et al.