

Re: Tuning Gigabit

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/performance/2003-06/0132.html>

From: David Gilbert (dgilbert_at_velocet.ca)

Date: 06/29/03

Date: Sat, 28 Jun 2003 22:23:17 -0400

To: Craig Reyenga <craig@craig.afraid.org>

>>>> "Craig" == Craig Reyenga <craig@craig.afraid.org> writes:

>> 300 megabit is about where 32bit 33Mhz PCI maxes out.

Craig> Could you tell me a little more about your tests? What boards,
Craig> and what configuration?

Well... first of all, a 33Mhz 32-bit PCI bus can transfer 33M * 32 bits ... which is just about 1 gigabit of _total_ PCI bus bandwidth. Consider that you're likely testing disk->RAM->NIC and you end up with 1/3 of that as throughput (minus bus overhead) so 300 megabit is a good number.

There are many ways boards can get around this. Your IDE controller can be on a different bus. Your RAM can be on a different bus. If all three are on different busses, you might get closer to your gigabit of throughput. You can also speed up the bus ... PCI can run at 66 or 100 Mhz. PCI-X can run at 66, 100 or 133 Mhz. You can also make the bus wider ... many new chipsets support 64 bit slots.

Now some boards I've tested (like the nvidia chipset) are strangely limited to 100megabit. I can't explain this. It seems low no matter how you cut it.

Our testing has been threefold:

- 1) Generating packets. We test the machines ability to generate both large (1500, 3000 and 9000 byte) and small (64 byte) packets. The large scale generation of packets is necessary for the other tests. So far, some packet flood utilities from the linux hacker camp are our most efficient small packet generators. netcat on memory cached objects or on /dev/zero generate our big packets.
- 2) Passing packets. Primarily, we're interested in routing. Passing packets, passing packets with 100k routes and passing packets with 100's of ipf accounting rules are our benchmarks. We look at both

freebsd-performance: Re: Tuning Gigabit

small and large packet performance. Packet passing machines have at least two interfaces ... but sometimes 3 or 4 are tested. Polling is a major win in the small packet passing race.

3) Receiving packets. netcat is our friend again here. Receiving packets doesn't appear to be the same level of challenge as generating or passing them.

At any rate, we're clearly not testing file delivery. We sometimes play with file delivery as a first test ... or for other testing reasons. We've found several boards that corrupt packets when they pass more than 100megabit of packets. We haven't explained that one yet. Our tests centre on routing packets (because that's what we do with our high performance FreeBSD boxes. All our other FreeBSD boxes "just work" at the level of performance they have).

Although I would note that we do have some strange datapoints where we've revisited old problems. One of the most peculiar is the DEC tulip chipset 4 port cards.

... on these cards ... we have only been able to ever pass 100megabit _per card_ ... never per port. It would appear that the PCI bridge on these cards is imposing some form of limitation. We haven't tested under any other OSs than FreeBSD ... but the problem is definitely perplexing.

Dave.

--

```
=====
|David Gilbert, Velocet Communications.      | Two things can only be      |
|Mail:          dgilbert@velocet.net       | equal if and only if they |
|http://daveq.ca                | are precisely opposite.   |
=====GLO=====
```

freebsd-performance@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-performance>

To unsubscribe, send any mail to "freebsd-performance-unsubscribe@freebsd.org"