HP WORLD 2004
Solutions and Technology Conference & Expo

Pump Up Your Network Server Performance with HP-UX

Paul Comstock

Network Performance Architect
Hewlett-Packard

© 2004 Hewlett-Packard Development Company, L.P.
The information contained herein is subject to change without notice

Purpose of this presentation

- Understand the factors affecting network performance, and what you can do about them
- Survey hardware and software options for HP-UX network servers
- Learn the network configuration and tuning parameters affecting performance

Benchmarking Tools

- SPEC benchmarks (www.specbench.org)
  - SPECweb99: static (70%) and dynamic (30%) HTTP
  - SPECweb99_SSL: w/SSL encryption/decryption
  - SPECweb2004: Under development – new workloads such as banking, e-commerce.
- Netperf (www.netperf.org)
  - Publicly available from HP
  - Measures maximum throughput (Stream) and transactional (Request-Response) performance
- Your application benchmark

Performance Tools

- HP-UX commands
  - ifconfig/lanscan/lanadmin
  - ndd
  - netstat (-s)
  - ping (for roundtrip time)
  - top
  - traceroute (for multi-hop networks)
- DSPP Developer Edge tools (www.hp.com)
  - vsar
  - caliper (for Itanium)
- HP-UX Internet Express (software.hp.com)
  - tcpdump
- Glanceplus (managementsoftware.hp.com)

Where is the bottleneck?

Bandwidth

Application

Protocol

Memory

CPU

Sample netstat –s output (partial)

```
->netstat -s
tcp:
        205723900 packets sent
                203496218 data packets (1453019982 bytes)
                107864 data packets (31506459 bytes) retransmitted
                2227182 ack-only packets (439786 delayed)
        100885096 packets received
                91622713 acks (for 1461278521 bytes)
                225582 duplicate acks
                14269401 packets (3611105775 bytes) received in-sequence
                4 completely duplicate packets (4346 bytes)
                435 packets with some dup, data (53746 bytes duped)
                5182 out of order packets (3064310 bytes)
                0 segments discarded for bad checksum
        241398 connection requests
        190879 connection accepts
        432277 connections established (including accepts)
        58200 retransmit timeouts
                6977 connections dropped by rexmit timeout
        0 connect requests dropped due to full queue
```

Increase your bandwidth

- Use 1 Gigabit Ethernet NICs instead of 100BT
- Use a NIC with offload features
- Trunk multiple interfaces using *HP Auto Port Aggregation* (APA) (software.hp.com)
- One of today’s CPUs can run a GigE link at full speed
- For scalability, use multiple NICs
- Spread device interrupts using *HP-UX Interrupt Migration* (software.hp.com)

Interrupt Migration – intctl command

```
# intctl

H/W Path class       drv     card cpu cpu intr   intr Card
                             name  cell   ID   cell type  ID  description
==========================================================================
0/0/0/0   lan      btlan   0    1    0    L     5    HP PCI10/100Base-TX Core
0/0/1/0   ext_bus  c720  0      1    0    L     0    SCSI C895 FastWide LVD
0/0/2/0   ext_bus  c720  0      2    0    L     1    SCSI C87x UltraWide Single-Ended
0/0/2/1   ext_bus  c720  0      3    0    L     2    SCSI C87x UltraWide Single-Ended
```

- Spread high speed network devices between CPUs
- Other devices, such as disks, may also be a concern depending on usage

Checksum Offload

TCP Segmentation Offload

Checksum Offload (CKO)

- Performs inbound and outbound TCP/UDP checksum calculations in hardware, offloading the host CPU
- Available for all HP-UX Gigabit Ethernet hardware
- Currently done for IPv4 only on HP-UX
- Example:

```
->ifconfig lan3
lan3: flags=1843<UP,BROADCAST,RUNNING,MULTICAST,CKO>
        inet 192.6.1.94 netmask ffffff00 broadcast 192.6.1.255
```

TCP Segmentation Offload (TSO)

IS:

- Segmentation of outbound data into IP datagrams in the NIC
- Required TCP/IP stack and NIC support
- Builds on CKO and offloads even more host processing
- Currently IPv4 only on HP-UX
- Uses a large virtual MTU (VMTU) internally, standard MTU on the wire

IS NOT:

- Not a new protocol on the wire
- Not jumbo frames

TSO Software

- Transport Optional Upgrade Release (TOUR) 2.2
- GigEtherEnh-01: Enhancement Software for GigEther-01
- Both are free from software.hp.com
- Configuration through lanadmin:

```
# lanadmin -x vmtu <ppa>
Driver/Hardware  supports  TCP  Segmentation
Offload. Current VMTU = 32160
```

New Offload Technologies

- Even more network processing may be offloaded in the future, as network speeds increase
- New technologies that provide network offload capability include RDMA, TCP Offload Engine (TOE), ETA, and iSCSI.
- These include TCP and non-TCP based technologies
- For more information, see break-out session "*What Is RDMA?*"

How much do offloads boost performance? “The answer is always ‘It depends’.”

A wise computer science instructor

Avoidance Maneuvers

Programming with Sendfile

```
sendfile(2)

 NAME
      sendfile() - send the contents of a file through a socket

 SYNOPSIS
      #include <sys/socket.h>

      sbsize_t sendfile(int s, int fd, off_t offset, bsize_t nbytes,
              const struct iovec *hdtrl, int flags);
```

- Sendfile avoids copying between file system and network buffers for TCP socket applications that send all or part of a file across the network
- Used by web servers (Zeus, Apache), and ftp on all versions of HP-UX

Network Server Accelerator

- NSA HTTP available for free from software.hp.com
- Uses a memory based cache to handle repetitive HTTP GET requests for static content
- Transparent to web server
- Avoids multiple socket system calls needed to accept a new connection and perform a web transaction
- Performance boost will vary depending on how much of the workload is static web requests.
- Limitations: doesn't help with dynamic or encrypted content
- For more info see break-out session *Accelerating Web Server Performance on HP-UX Using NSA HTTP*

Configuration and Tuning

Network Stack Configuration

- A number of network tunables are commonly modified on big servers or in high performance environments
- tcphashsz (system tunable) default 2048; tune up to 64K for large configurations
- tcp_conn_request (ndd tunable) default 4096; good in most cases; be sure to use a large backlog when calling listen(2)
- socket_caching_tcp (ndd_tunable) default 1 (on); use a number greater than 512 based on number of simultaneous TCP connections in use
- SO_SENDBUF/SO_RCVBUF (setsockopt(2)) default 32768; SO_RCVBUF sets the TCP receive window; SO_SENDBUF helps determine when outbound flow control occurs

Determining the Receive Buffer

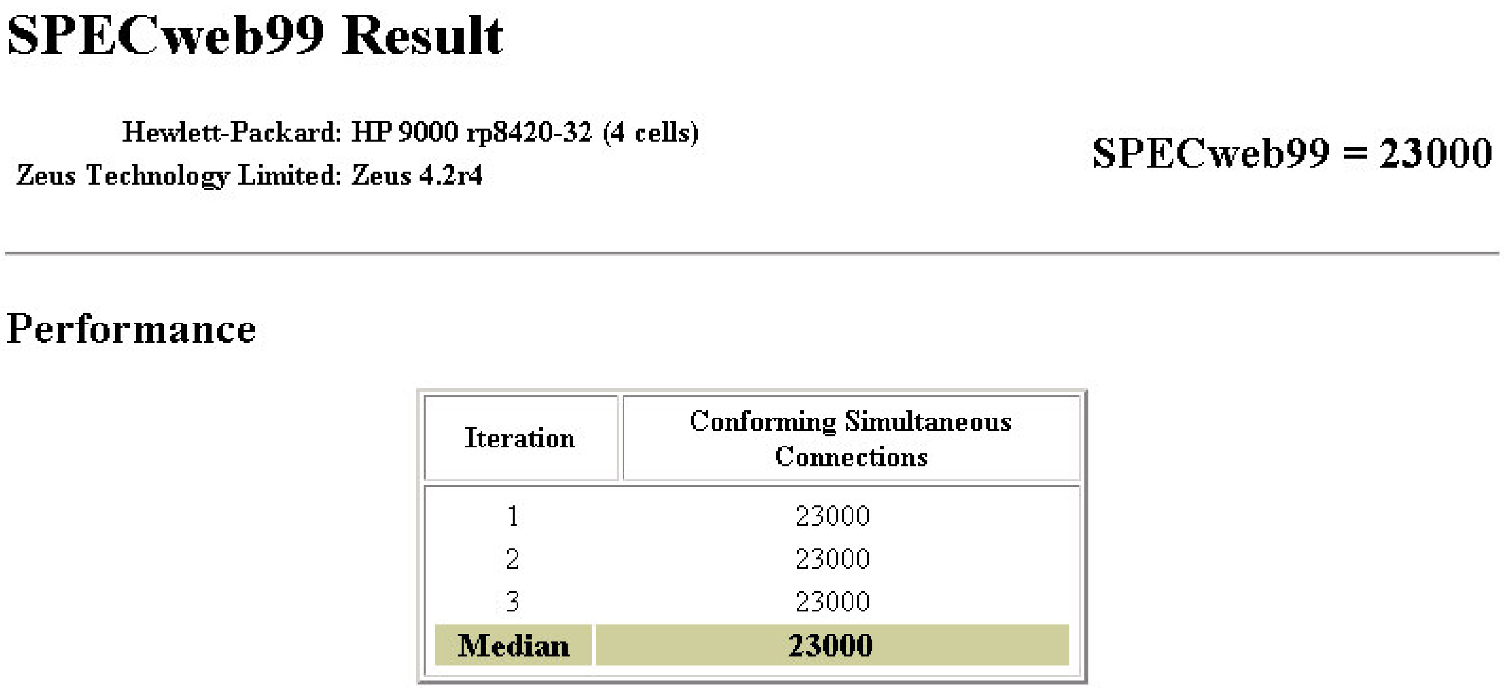
- For long, fat pipes (LFPs), a large receive buffer may be needed to use all of the available bandwidth.
- LFPs have a long round trip time (RTT), and high (fat) bandwidth, so lots of data can be in transit
- The minimum buffer can be determined by the formula rcvbuf = RTT * BW
- RTT can be determined with ping, or more accurately on actual TCP connections using tcpdump
- For example, on a 100 Mbit network has a 80 ms round trip time. The rcvbuf should be 100,000,000 b/s * .08 s = 8 Mbits (1 MB)

Parameters for Networks with Special Needs

- TCP Selective Acknowledgement (SACK)
  - RFC 2018, uses option fields in TCP header
  - Faster retransmission of multiple gaps in sequence space
  - tcp_sack_enable (ndd) default 2 (don’t initiate SACK)
- tcp_smoothed_rtt (ndd) default 0; can be used for networks with volatile delay behavior such as those with satellite-based and cellular links
- tcp_rexmit_interval_min/tcp_rexmit_interval_max (ndd) default 500 ms/60 sec; not usually changed, as timer-based retransmissions are not that common, and the actual interval is based on RTT measurements
- TCP_NODELAY (setsockopt) default 0; avoids delays in transmission of small segments (Nagle algorithm), but won’t help system-wide performance

Anatomy of a SPECweb99 Result

- How to read a SPECweb disclosure
- Examples of tuning parameters from an actual benchmark



SPECweb99 Result

Hewlett-Packard: HP 9000 rp8420-32 (4 cells)
Zeus Technology Limited: Zeus 4.2r4

SPECweb99 = 23000

Performance

Iteration	Conforming Simultaneous Connections
1	23000
2	23000
3	23000
Median	**23000**

http://www.specbench.org/web99/results/res2004q1/web99-20040211-00259.html

References

- Transport Optional Upgrade Release (TOUR) 2.0 FAQ *(HP-UX 11i v1, HP-UX 11i v2)*, docs.hp.com/hpux/netcom
- Network Server Accelerator HTTP PerformanceWhite Paper *(HP-UX 11i v1)*, docs.hp.com/hpux/internet
- PCI-X 2 Gigabit Fibre Channel and Gigabit Ethernet Performance Paper *(HP-UX 11i v1, HP-UX 11i v2)*, docs.hp.com/hpux/netcom
- PCI Gigabit Ethernet Performance on HP Server rp7410 *(HP-UX 11i v1)* , docs.hp.com/hpux/netcom
- Using APA to Build a Screaming Fast Network Server Connection, docs.hp.com/hpux/netcom
- Running SPECweb99 with Zeus, Zeus Technology, http://support.zeus.com/doc/tech/SPECweb99.pdf
- Web Servers for HP-UX, http://www.hp.com/products1/unix/webservers

invent

HP WORLD 2004

Solutions and Technology Conference & Expo

Co-produced by:

interex

shared knowledge • shared power

encompass

AN HP USER GROUP

RECOMMENDED TRAINING VENUE FOR THE

HP Certified Professional