mission critical linux

Freedom - Innovation - Confidence

Linux Server Performance Notes

This short paper will focus on identifying some general server configuration issues which might affect performance, and will provide configuration and tuning suggestions for increasing the overall performance characteristics of a Linux Server. In addition to tuning suggestions, we will also include a list of links to sites which were consulted in the creation of this paper, and which discuss Linux system performance issues in general.

Tuning servers for optimal performance is a difficult task, dependent on many variables:

- Hardware configuration
- Software configuration
- Tunable system parameters
- Supported application behavior
- Performance analysis & benchmarks

This paper will address the first three issues, but the behavior of applications on a specific server, including the actual measurement of that behavior, is difficult to address in a generic manner. When looking at performance in such a specific way, individual environment analysis, measurements, and system experimentation will be required to optimize system and application performance on a server.

In order to provide an initial list of specific suggestions, Mission Critical Linux has reviewed some relevant performance benchmark results published by several vendors. The two primary benchmarks reviewed are managed by the Standard Performance Evaluation Corporation (http://www.spec.org/), an independent consortium created to provide reliable, consistent, and fair benchmarks for all software/hardware vendors, and to assure that benchmark data published by all vendors meets strict criteria assuring its reliability. We did not look at the TPC database benchmarks (http://www.tpc.org/), primarily because no vendor has published results obtained from Linux based servers. The two benchmarks consulted are:

- SPECsfs97(r1) – The most recently released NFS server benchmark (http://www.spec.org/osg/sfs97r1/), measuring the capabilities of an NFS server to handle a typical server load, based on customer analysis from the NFS vendors. This benchmark has two components, which measure the two NFS protocols – V2 and V3.

- SPECweb99 – The latest World Wide Web server benchmark (http://www.spec.org/osg/web99/), also presenting servers with a standard workload generated from socket-based network traffic, and designed to run on all servers running Unix, Linux, and Microsoft NT operating systems.

In reviewing the benchmark data, and the configurations used by the vendors to generate their results, we concentrated on both single and dual processor systems, since this configuration seems to be more commonly used than any other. We also concentrated on Linux based servers for the SPECweb benchmark, but needed to use Unix systems in the NFS benchmark review, since no data has been published for Linux based platforms. Since the relevant system configuration issues for NFS should be similar for all Unix and Linux platforms, this did not seem to present a significant problem.

The Servers reviewed:

Vendor	Platform	# CPUs	OS	Benchmark
Compaq	Proliant DL320	1	Red Hat Linux 7.0	SPECweb99
Compaq	Proliant ML370G2	2	Red Hat Linux 7.1	SPECweb99
Dell	PowerEdge 2500	2	Red Hat Linux 7.1	SPECweb99
Dell	PowerEdge 4400/800	2	Red Hat Linux 6.2	SPECweb99
IBM	eServer xSeries 370	1	Red Hat Linux 7.0	SPECweb99
IBM	eServer xSeries 370	2	Red Hat Linux 7.0	SPECweb99
IBM	Netfinity 8500R	2	Red Hat Linux 7.0	SPECweb99
HP	hp server rx4610	4	HP-UX 11.20	SPECsfs97
Network Appliance	F820c Cluster	2	Data OnTap 6.1.1	SPECsfs97
Network Appliance	F840c Cluster	2	Data OnTap 6.1.1	SPECsfs97

Hardware Configuration

In addition to the benchmarks, the currently published reviews of Linux performance factors were reviewed as well. Most of this data is available on the web, and will be listed at the end of this document, as mentioned above. Based on this review, the following list of system components is meant to identify the factors which should be considered when determining modifications that will have the most effect on system performance. This is a general list, formed by inspection of several hardware platforms, and since each specific system platform may have unique components, this list should not be considered as definitive.

- Processor boards
  - 2-way CPU
  - 32 Kbytes level 1 cache (on-processor memory)
  - 256 Kbytes level 2 cache (off-processor memory)

- Memory
  - 2 Gbytes RAM

- Disk Controllers
  - Multiple controllers
  - SCSI or FibreChannel

- File Systems
  - Type dependent on need (see note #1)
  - Raw IO partition for swap space (do not use FS swap partitions)
  - Separate partition for /var, or /var/log, plus /var/spool for mail servers
  - 4K file block size

- Network
  - Multiple Gigabit Ethernet NICs
  - Full Duplex network configuration (see note #2)
  - TCP Network Protocol

Note #1 – The file system selected should match the needs of the applications run on this server. If the reliability of a journaled file system is required, ReiserFS is supported by Convolo DataGuard release 2.0.

Note #2 – Autoconfigured NICs do not guarantee full duplex behavior if switches, routers, or other devices in the subnet are fixed at half duplex. Be sure to force all devices to use full duplex if at all possible.

Software Configuration

Several software configuration changes should be considered when initially setting up the system software, including several modifications executed by system commands, as well as options taken when configuring subsystems such as NFS. Based on the benchmark results cited above, as well as our own benchmark investigations and some suggestions given in the performance discussions cited at the end of this paper, we think the following modifications are helpful.

- Kernel
  - 2.4.* kernels provide better file system performance, disk access performance, and overall NFS performance

- File System
  - Increase the number of open files allowed with the ulimit command (note that the –a option will give the current settings for all parameters): *ulimit –n 2048*

- Network
  - TCP protocol

- NFS
  - Use the *-o vers=3* option in the client mount command to select NFS protocol version 3.
  - Use the *–o noatime* option to turn off unused access time updates
  - use the *–o wsize=8192,rsize=8192* option to set the NFS read and write transfer size to the current maximum.

System Parameter Tuning

Linux system parameters can be tuned in several ways, depending on your system platform. All distributions can be tuned through modifying values of writeable files in the /proc file system with the 'echo' command. To preserve these modifications, the alterations should be inserted into a system configuration file; on most systems the changes can be kept in /etc/rc.d/rc.local. For later Red Hat distributions, system parameter values can be saved in a specific system configuration file, /etc/sysctl.conf.

In order to modify the maximum number of file handles allocated by the kernel, for example, the following command would be executed by a user having root privileges:

echo "32768" > /proc/sys/fs/file-max

When inserting this type of alteration into the /etc/sysctl.conf directory, on Red Hat 6.2 and later releases, the variable name is altered slightly:

fs.file-max = 32768

Documentation taken from the 2.4.6 Linux source tree, giving an explanation of many system variables such as the above, will be attached to this document. This documentation can be found in */usr/src/linux-2.4.7/Documentation/sysctl*, as well as */usr/src/linux-2.4.7/Documentation/networking*.

Based on the needs of the application, and on the requirements of the environment, system parameters should be tuned to deliver the highest performance in that particular situation. These values should be considered to be extremely dynamic, and should be documented when not using the system defaults . Some suggested configuration values will be given below, based on our inspection of published benchmark data, and our review of performance discussions published on the web, but these should be considered as a starting point for learning their effect, rather than a fixed recommendation.

Parameter	Meaning	Possible Value
/proc/sys/fs/file-max (fs.file-max)	Maximum number of allocated file handles	16384
/proc/sys/net/core/wmem_max (net.core.wmem_max)	Maximum size of send socket buffer	1048576
/proc/sys/net/core/wmem_default	Default size of send socket buffer	131072
/proc/sys/net/core/rmem_max	Maximum size of receive socket buffer (see note #1)	1048576
/proc/sys/net/core/rmem_default	Default size of receive socket buffer	131072
/proc/sys/net/core/optmem_max	Maximum number of optional memory buffers	20480
/proc/sys/net/core/hot_list_length	Maximum number of skbuffer heads	10000
/proc/sys/net/ipv4/tcp_timestamps (net.ipv4.tcp_timestamps)	Timestamp support	0
/proc/sys/net/ipv4/tcp_wmem	TCP receive buffer space	131072
/proc/sys/net/ipv4/tcp_rmem	TCP send buffer space	131072
/proc/sys/net/ipv4/tcp_max_tw_buckets	Maximum size of TCP time wait bucket pool	8192
/proc/sys/vm/bdflush	Bdflush kernel daemon behavior	See note #2
/proc/sys/vm/freepages	Kernel swapping behavor	See note #2

Note #1 – The maximum and default socket buffer sizes will depend on the NIC speed; The maximum number given assumes gigabit speed.

Note #2 – The vm parameters have multiple values; see the documentation text cited at the end of this document for complete details. The value suggested by the benchmark data is as follows:

echo “100 5000 640 2560 150 30000 5000 1884 2” > /proc/sys/vm/bdflush

In addition, a discussion of these values can be found in the BYTE Magazine column cited below; see page 3 (of 5) in that column. It cites the meaning of the first three parameters:

nfract – The number of dirty buffers in the buffer cache
ndirty – The maximum number of dirty buffers written to the disk in one operation
nrefill – The number of free buffers allocated

Additionally, */proc/sys/vm/freepages* will control kernel swapping behavior. In the same column in BYTE magazine, the author states the meaning of the three parameters:

min – The minimum number of free pages allowed to non-kernel processes
low – The minimum number of free pages required to prevent high-priority swapping
high – The minimum number of free pages required to prevent low priority swapping

Finally, the use of the vm parameters found in the /proc file system in the 2.4 kernels is limited. For these kernels, you should review the use of vm in the 2.4 kernel documentation. In addition, the Linux kernel mailing list has discussions concerning vm management, as well as system performance management in general. Reviewing these discussions is encouraged.

Summary

As mentioned, when looking at performance in detail, and when wanting to maximize performance optimizations, individual environment analysis, measurements, and system experimentation will be required. Mission Critical Linux is ready to assist any customer with this analysis, and will provide recommendations for configuration and tuning in order to deliver optimal performance characteristics, both for the server in general, and for specific server applications which are of greatest interest to the customer.

mission critical linux

Freedom - Innovation - Confidence

Appendix 1 - Information Sources

The table below gives a list of hyperlinks to several interesting and informative pages dealing with performance tuning in the Linux operating system. The first two links contain tuning discussions, as well as long lists of relevant pointers to further suggestions, discussions, and tools. Following that are several articles which are informative in general, and the table ends with pointers to two performance monitoring tools.

Performance Tuning Hyperlink	Contents
http://tunelinux.com/	Linux performance tuning gateway page
http://linuxperf.nl.linux.org/	Linux performance tuning discussions
http://www.colltech.com/real_world_solutions/bottlenecks.html	Performance tuning white paper
http://people.redhat.com/alikins/system_tuning.html	Red Hat server tuning review
http://www.networkcomputing.com/1122/1122ws2.html	Network Computing Magazine performance tuning paper (11/00)
http://www.byte.com/documents/s=429/byt20000829s0006/	Byte Magazine column on Linux performance tuning (8/00)
http://www.psc.edu/networking/perf_tune.html	Pittsburgh Supercomputing Center performance review
http://www.linux.com/enhance/tuneup/	OSDN “Tips & Tweaks” database
http://www.kegel.com/	General performance discussions
http://www.citi.umich.edu/projects/	The Univ. of Michigan’s Center for Information Technology Integration
http://www.tpc.org/	The Transaction Processing Performance Council (TPC-C, etc.)
http://www.spec.org/	The Standard Performance Evaluation Corporation (Specweb99, etc.)
http://www.blakeley.com/resources/vtad/vtad-pod.html	“Rule Based” Performance monitoring tool (download site)
http://tunelinux.com/bin/page?daemons/nfs/nfspmon.html	Source code for a performance monitoring tool

Appendix 2 – Source Code Documentation

Reviews of system parameters is contained in the Linux source documentation for the 2.4.7 Linux kernel, in the following files (this assumes that the Linux source tree resides in /usr/src/linux):

/usr/src/linux/Documentation/sysctl/fs.txt
/usr/src/linux/Documentation/sysctl/kernel.txt
/usr/src/linux7/Documentation/sysctl/sunrpc.txt
/usr/src/linux/Documentation/sysctl/vm.txt
/usr/src/linux/Documentation/networking/ip-sysctl.txt