

Re: Tuning Postgresql on FreeBSD 5.1

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/performance/2003-08/0025.html>

From: Paul Pathiakis (ppathiakis_at_micropat.com)

Date: 08/26/03

To: paul@pathiakis.com, freebsd-database@freebsd.org, freebsd-performance@freebsd.org
Date: Tue, 26 Aug 2003 14:20:16 -0400

Ok....everyone wanted more info so....

I've attached the ipcs -a log for the run
I've attached the vmstat log for the run

(I wanted the iostat, but screwed that up.)

5 large processes were kicked off. The processes started off ok but then just stalled. They continued to run until completion but VERY slowly. Many times I saw the processes in semwait states. However, the CPU load never went over 0.80. Everything just kind of stalled and waited. They consumed huge amounts of memory but never swapped.

Again, thanks to everyone for taking the time!!

Paul Pathiakis

I disabled HTT in the BIOS, but the OS still saw 4 cpus... I'm going to again try to disable HTT and see what occurs in the kernel boot. For right now, I've enable and modified all the parameters according to what Sean suggested. I'm hoping they've turned up something here. In the event that somehow I can't disable, I'll use the cpu sysctl parameter that was suggested.

The stats from top are from 5 processes that were run simultaneously:

last pid: 8704; load averages: 0.10, 0.15, 0.10 up 0+04:26:55
14:22:37

52 processes: 1 running, 51 sleeping

CPU states: 2.4% user, 0.0% nice, 0.9% system, 0.6% interrupt, 96.1% idle

Mem: 181M Active, 3185M Inact, 296M Wired, 187M Cache, 255M Buf, 5516K Free

Swap: 4096M Total, 4096M Free

```
PID USERNAME PRI NICE SIZE RES STATE C TIME WCPU CPU COMMAND
8171 postgres -8 0 129M 116M biord 0 1:01 5.32% 5.32% postgres
8173 postgres -8 0 137M 125M biord 0 0:43 3.76% 3.76% postgres
8320 postgres 96 0 2188K 1176K select 2 0:01 0.00% 0.00% top
```

freebsd-performance: Re: Tuning Postgresql on FreeBSD 5.1

```
598 pgsq1 96 0 6164K 2232K select 2 0:00 0.00% 0.00% sshd
652 pgsq1 96 0 48696K 4160K select 0 0:00 0.00% 0.00% postgres
599 pgsq1 8 0 2556K 1664K wait 0 0:00 0.00% 0.00% ksh93
655 pgsq1 96 0 48748K 2976K select 0 0:00 0.00% 0.00% postgres
653 pgsq1 96 0 49684K 2944K select 0 0:00 0.00% 0.00% postgres
```

On Monday 25 August 2003 09:29 am, Paul Pathiakis wrote:

> Hi,
>
> I'd like to thank everyone for their help so far. I'm implement most of
> the enhancements and changes on the database so far. I now have a bigger
> problem... the machine is generating some large reports (term used by the
> DB people here) and the processes start and instantly drop off to no
> utilization... they sit there and hang... seemingly resource starved.
> I'd like know if someone could help me. I've enclosed the postgresql.conf
> file, the systctl.conf file, loader.conf and the Kernel memory parameters.
> Again, the machine is a twin 2.8 Xeon HTT machine. HTT is turned on and
> the machine sees 4 cpus. It has 4 GB of RAM and I'm starting to put on
> SCSI drives as the machine had the IDE drives maxed out at 100% utilization
> at all times. (Again, I didn't order this machine, otherwise it would have
> had dual U320 channels etc on its I/O system) Presently, due to scavenging
> a PCI SCSI card (ADAPTEC U2W), disks (U320 10K rpm) and an enclosure for 4
> disks, I have the following:
>
> /dev/ar1s1d 114244630 479980 104625080 0% /usr/local
> /dev/da0s1d 138860928 81448860 46303194 64% /usr/local/pgsql
> /dev/md0 1031916 4 949360 0%
> /usr/local/pgsql/data/base/16978/pgsql_tmp
> procfs 4 4 0 100% /proc
> /dev/da1s1d 138860928 131202 127620852 0%
> /usr/local/pgsql/data/pg_xlog
> /dev/da2s1d 142801720 3277472 128100112 2% /pg_index
> /dev/da3s1d 142801720 1049616 130327968 1% /pg_table
>
> noatime is configured on the pgsq1 hierarchy and related links to disks on
> /pg_*. I configured a MD as the pgsq1_tmp directory, etc. I've created
> UFS2+S filesystems with block and frag sizes of 8K as this is optimal for
> PG. I hope this makes sense all the way around. I'm not a DBA, just a UNIX
> admin.
>
> Anyhow, I'm looking at the postgresql.conf file and I don't see a whole
> lot that makes sense to me. Please help! I've got a bunch of people
> saying Linux just runs faster and the DB group is using a Linux config file
> on the FreeBSD machine. (Don't get me going) It is my belief that a BSD
> DB is going to run faster on it's platform of choice for development
> (FreeBSD) than another OS. Please help!
>
> Thanks!
>
> Paul Pathiakis
>

freebsd-performance: Re: Tuning Postgresql on FreeBSD 5.1

```
>
> sysctl.conf:
>
> kern.maxfiles=10000
> kern.ipc.shm_use_phys=1
> kern.ipc.shmall=524288
> kern.ipc.shmmax=1073741824
> vfs.vmiodirenable=1
>
> loader.conf
>
> kern.maxfiles=32768
> kern.nbuf=16384
>
> GENERIC SMP:
>
> options SYSVMSG #SYSV-style message queues
> options SYSVSHM #SYSV-style shared memory
> options SYSVSEM #SYSV-style semaphores
> options SHMMAXPGS=4096
> options SHMSEG=256
> options SEMMNI=256
> options SEMMNS=512
> options SEMMNU=256
> options SEMMAP=256
>
> Postgresql.conf:
>
> #
> # PostgreSQL configuration file
> # -----
> #
> # This file consists of lines of the form:
> #
> # name = value
> #
> # (The '=' is optional.) White space may be used. Comments are introduced
> # with '#' anywhere on a line. The complete list of option names and
> # allowed values can be found in the PostgreSQL documentation. The
> # commented-out settings shown in this file represent the default values.
> #
> # Any option can also be given as a command line switch to the
> # postmaster, e.g. 'postmaster -c log_connections=on'. Some options
> # can be changed at run-time with the 'SET' SQL command.
> #
> # This file is read on postmaster startup and when the postmaster
> # receives a SIGHUP. If you edit the file on a running system, you have
> # to SIGHUP the postmaster for the changes to take effect, or use
> # "pg_ctl reload".
>
>
```

freebsd-performance: Re: Tuning Postgresql on FreeBSD 5.1

```
>
#=====
>
>
> #
> # Connection Parameters
> #
> #tcpip_socket = false
> #ssl = false
>
> #max_connections = 128
> #superuser_reserved_connections = 2
>
> #port = 5432
> #hostname_lookup = false
> #show_source_port = false
>
> #unix_socket_directory = ""
> #unix_socket_group = ""
> #unix_socket_permissions = 0777 # octal
>
> #virtual_host = ""
>
> #krb_server_keyfile = ""
>
>
> #
> # Shared Memory Size
> #
> #shared_buffers = 48000 # min max_connections*2 or 16, 8KB each
> #max_fsm_relations = 1000 # min 10, fsm is free space map, ~40 bytes
> #max_fsm_pages = 10000 # min 1000, fsm is free space map, ~6 bytes
> #max_locks_per_transaction = 64 # min 10
> #wal_buffers = 8 # min 4, typically 8KB each
>
> #
> # Non-shared Memory Sizes
> #
> #sort_mem = 32768 # min 64, size in KB
> #vacuum_mem = 8192 # min 1024, size in KB
>
>
> #
> # Write-ahead log (WAL)
> #
> #checkpoint_segments = 3 # in logfile segments, min 1, 16MB each
> #checkpoint_timeout = 300 # range 30-3600, in seconds
> #
> #commit_delay = 0 # range 0-100000, in microseconds
> #commit_siblings = 5 # range 1-1000
> #
```

freebsd-performance: Re: Tuning Postgresql on FreeBSD 5.1

```
> #fsync = true
> #wal_sync_method = fsync # the default varies across platforms:
> # # fsync, fdatasync, open_sync, or
> open_datasync
> #wal_debug = 0 # range 0-16
>
>
> #
> # Optimizer Parameters
> #
> #enable_seqscan = true
> #enable_indexscan = true
> #enable_tidscan = true
> #enable_sort = true
> #enable_nestloop = true
> #enable_mergejoin = true
> #enable_hashjoin = true
>
> #effective_cache_size = 1000 # typically 8KB each
> #random_page_cost = 4 # units are one sequential page fetch cost
> #cpu_tuple_cost = 0.01 # (same)
> #cpu_index_tuple_cost = 0.001 # (same)
> #cpu_operator_cost = 0.0025 # (same)
>
> #default_statistics_target = 10 # range 1-1000
>
> #
> # GEQO Optimizer Parameters
> #
> #geqo = true
> #geqo_selection_bias = 2.0 # range 1.5-2.0
> #geqo_threshold = 11
> #geqo_pool_size = 0 # default based on tables in statement,
> # range 128-1024
> #geqo_effort = 1
> #geqo_generations = 0
> #geqo_random_seed = -1 # auto-compute seed
>
>
> #
> # Message display
> #
> #server_min_messages = notice # Values, in order of decreasing detail:
> # debug5, debug4, debug3, debug2, debug1,
> # info, notice, warning, error, log,
> # fatal, # panic
> #client_min_messages = notice # Values, in order of decreasing detail:
> # debug5, debug4, debug3, debug2, debug1,
> # log, info, notice, warning, error
> #silent_mode = false
>
```

freebsd-performance: Re: Tuning Postgresql on FreeBSD 5.1

```
> #log_connections = false
> #log_pid = false
> #log_statement = false
> #log_duration = false
> #log_timestamp = false
>
> #log_min_error_statement = panic # Values in order of increasing severity:
> # debug5, debug4, debug3, debug2,
> debug1, # info, notice, warning, error, panic(off)
>
> #debug_print_parse = false
> #debug_print_rewritten = false
> #debug_print_plan = false
> #debug_pretty_print = false
>
> #explain_pretty_print = true
>
> # requires USE_ASSERT_CHECKING
> #debug_assertions = true
>
>
> #
> # Syslog
> #
> #syslog = 0 # range 0-2
> #syslog_facility = 'LOCAL0'
> #syslog_ident = 'postgres'
>
>
> #
> # Statistics
> #
> #show_parser_stats = false
> #show_planner_stats = false
> #show_executor_stats = false
> #show_statement_stats = false
>
> # requires BTREE_BUILD_STATS
> #show_btree_build_stats = false
>
>
> #
> # Access statistics collection
> #
> #stats_start_collector = true
> #stats_reset_on_server_start = true
> #stats_command_string = false
> #stats_row_level = false
> #stats_block_level = false
>
>
```


freebsd-performance: Re: Tuning Postgresql on FreeBSD 5.1

- > *freebsd-database@freebsd.org mailing list*
- > <http://lists.freebsd.org/mailman/listinfo/freebsd-database>
- > *To unsubscribe, send any mail to "freebsd-database-unsubscribe@freebsd.org"*

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"

- text/plain attachment: [ipcspg04](#)
-

- text/plain attachment: [vmstatpg04](#)