

Chapter 14 I/O Monitoring and Tuning - Notes

14.3 Learning Objectives:

- Understand the importance of monitoring I/O activity and when it constitutes system performance bottlenecks.
- Use **iostat** to monitor system I/O device activity.
- Use **iotop** to display a constantly updated table of current I/O usage.
- Use **ionice** to set both the **I/O scheduling class** and the **priority** for a given process.

14.4 I/O Monitoring and Disk Bottlenecks

Disk performance problems -> strongly coupled to other factors, eg. insufficient memory, inadequate network hardware/tuning. Disentanglement difficult.

Rule: system considered as **I/O-bound** when CPU found sitting idle waiting for I/O to complete, or network waiting to clear buffers.

However, can be misled. What appears to be insufficient memory can result from too slow I/O. If memory buffers being used for reading/writing fill up, may appear that memory is problem, when real problem is that buffers are not filling up or emptying out fast enough. Similarly, network transfers may be waiting for I/O to complete, causing network throughput to suffer.

Real-time monitoring + tracing -> both necessary tools for locating/mitigating disk bottlenecks. However, rare/non-repeating problems can make this difficult to accomplish.

Many relevant variables, I/O tuning complex. Will also consider **I/O scheduling** later.

14.5 iostat

iostat: basic workhorse utility for monitoring I/O device activity on system. Can generate reports with lot of information, with precise content controlled by options.

Can see below what simply typing **iostat** shows:

```
File Edit View Search Terminal Help
c7:/tmp>iostat
Linux 4.11.3 (c7)          06/02/2017          _x86_64_          (8 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
            1.27    0.00    0.94    0.25    0.00   97.53

Device:            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
sda                 0.60         32.32         0.25     509530      3908
sdb                11.76        283.35        176.27    4467320    2779108
dm-0                 0.19         30.42         0.24     479617      3812
dm-1                 0.07          0.33          0.00        5196         12
dm-2                 0.02          0.12          0.00        1921         12
dm-3                 0.03          0.15          0.00        2417         12
dm-4                 0.02          0.12          0.00        1892         48
...
dm-12                1.59        118.88        74.54    1874200    1175172
loop0                0.99          1.01          0.00        16002          0

coop@c7:/tmp
c7:/tmp>
```

After brief summary of CPU utilization, I/O statistics given: **tps** (I/O transactions per second; logical requests can be **merged** into one actual request), clocks read/w ritten per unit time, w here blocks are generally sectors of 512 bytes; total blocks read/w ritten.

Information broken out by disk partition (and if LVM is being used also by **dm**, or device mapper, logical partitions).

14.6 iostat Options

Somew hat different display generated by giving **-k** option, results in showing KB instead of blocks. Can also use **-m** to get results in MB.

```
$ iostat -k
```

```
File Edit View Search Terminal Help
c7:/tmp>iostat -k
Linux 4.11.3 (c7)          06/02/2017          _x86_64_          (8 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
            1.30    0.00    0.94    0.27    0.00   97.48

Device:            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
sda                 0.82        258.31         0.25    4211894      4064
sdb                11.96        274.27        174.60    4472104    2846868
dm-0                 0.19         30.67         0.24     500021      3944
dm-1                 0.07          0.32          0.00        5196         12
dm-2                 0.02          0.12          0.00        1921         12
dm-3                 0.03          0.15          0.00        2417         12
dm-4                 0.24        225.93         0.00    3683852         72
...
dm-12                1.55        114.94        72.17    1874200    1176720
loop0                0.96          0.98          0.00        16002          0
c7:/tmp>
c7:/tmp>
```

Another useful option: **-N**, to show device name (or **-d** for somew hat different format), as show n below :

```
$ iostat -N
```

```
File Edit View Search Terminal Help
c7:/tmp>iostat -N
Linux 4.11.3 (c7)           06/02/2017           _x86_64_           (8 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
            1.29    0.00    0.94    0.27    0.00   97.49

Device:            tps    kB_read/s    kB_wrtn/s    kB_read  kB_wrtn
sda                 0.81       256.52        0.25     4219286     4108
sdb                12.05       272.67       174.42    4484900    2868920
VG2-pictures        0.19        30.85         0.24     507413     3988
VG2-dead            0.07         0.32         0.00        5196         12
VG2-iso_images      0.02         0.12         0.00       1921         12
VG2-audio           0.03         0.15         0.00       2417         12
VG2-virtual         0.24      223.97         0.00    3683852        72
VG2-w7back          0.00         0.03         0.00        456          0
VG2-P               0.00         0.00         0.00         76          0
VG2-isabelle        0.00         0.03         0.00        456          0
VG2-dead2           0.21         0.88         0.00     14489         12
VG2-PLAY            0.00         0.03         0.00        456          0
VG-local            7.02       35.91       65.06    590629    1070180
VG-src              1.27         5.85         0.44     96264       7224
VG-vms              1.54      113.95       71.56   1874200   1177052
loop0               0.95         0.97         0.00     16002          0
c7:/tmp>
c7:/tmp>|
```

14.7 iostat Extended Options

Much more detailed report obtained by using `-x` option (for extended).

```
$ iostat -xk
```

```
File Edit View Search Terminal Help
c7:/tmp>iostat -xk
Linux 4.11.3 (c7)           06/02/2017           _x86_64_           (8 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
            1.30    0.00    0.94    0.27    0.00   97.49

Device:            rrqm/s   wrqm/s     r/s     w/s    kB/s    kB/s   avgrq-sz   avgrq-sz   await   r_await   w_await   svctm   %util
sda                 0.00     0.02   0.72   0.08   251.52    0.25   630.23    0.05   63.63   58.61   109.68   25.59   2.04
sdb                 0.09     4.66   5.79   6.41   267.04   174.83    72.47    0.03    2.41    0.49    4.14    0.43   0.52
dm-0                 0.00     0.00   0.13   0.06    30.87    0.24   330.18    0.02  128.02  126.65   130.89   97.60   1.84
dm-1                 0.00     0.00   0.07   0.00     0.31    0.00     9.36    0.00    7.93    7.64    73.20    4.19   0.03
dm-2                 0.00     0.00   0.02   0.00     0.11    0.00    14.01    0.00   15.46   14.99    41.40   14.12   0.02
dm-3                 0.00     0.00   0.03   0.00     0.14    0.00    11.09    0.00   36.09   36.28   19.80   12.95   0.03
dm-4                 0.00     0.00   0.23   0.00   218.98    0.00  1897.46    0.02   95.29   94.75   199.20    8.82   0.20
dm-12                0.00     0.00   1.12   0.38   111.41   70.02   240.59    0.02   11.01    0.86    40.76    0.44   0.07
loop0               0.00     0.00   0.93   0.00     0.95    0.00     2.05    0.00    0.01    0.01    0.00    0.01   0.00
c7:/tmp>
c7:/tmp>|
```

Fields seen above have following meanings:

Extended iostat Fields

Field	Meaning
Device	Device or partition name

rrqm/s	Number of read requests merged per second, queued to device
wrqm/s	Number of write requests merged per second, queued to device
r/s	number of read requests per second, issued to device
w/s	number of write requests per second, issued to device
rkB/s	KB read from the device per second
wkB/s	KB written to the device per second
avgrq-sz	Average request size in 512 byte sectors per second
avgqu-sz	Average queue length of requests issued to the device
await	Average time (in msecs) I/O requests between when a request is issued and when it is completed: queue time plus service time
svctm	Average service time (in msecs) for I/O requests
%util	Percentage of CPU time during the device serviced requests

Note: if utilization percentages approaches 100, system saturated, or I/O bound.

14.8 iotop

iotop -> another very useful utility, must be run as root. Displays table of current I/O usage, updated periodically, like **top**. Can see below what typing `sudo iotop` with no options shows.

Note: **be** and **rt** entries in **PRI0** field explained in **ionice** section, stand for **best effort** and **real time**.

File Edit View Search Terminal Help							
Total DISK READ :		116.67 M/s	Total DISK WRITE :		132.23 K/s		
Actual DISK READ:		116.67 M/s	Actual DISK WRITE:		0.00 B/s		
TID	PRI0	USER	DISK READ	DISK WRITE	SWAPIN	IO>	COMMAND
17932	be/4	root	0.00 B/s	0.00 B/s	0.00 %	99.99 %	[kworker/2:0]
3571	be/4	coop	0.00 B/s	0.00 B/s	0.00 %	99.99 %	skype-bin
15601	be/4	coop	0.00 B/s	81.67 K/s	0.00 %	99.99 %	vmware-vmx -ssnapshot.num-ntu-17-04.vmx [vmx-vcpu-3]
19800	be/4	coop	0.00 B/s	0.00 B/s	0.00 %	99.99 %	gnome-screenshot -i -w
17186	be/4	coop	0.00 B/s	46.67 K/s	0.00 %	0.00 %	vmware-vmx -ssnapshot.num-17-04.vmx [vmx-vthread-16]
15598	be/4	coop	0.00 B/s	3.89 K/s	0.00 %	0.00 %	vmware-vmx -ssnapshot.num-ntu-17-04.vmx [vmx-vcpu-0]
19782	be/4	root	116.67 M/s	0.00 B/s	0.00 %	0.00 %	cat ./VIRTUAL/FC-25-LATEX/FC-25.vmdk
1	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	systemd --switched-root --system --deserialize 21
2	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kthreadd]
4	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kworker/0:0H]
6	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[mm_percpu_wq]
7	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/0]

Available options shown by using `--help` option.

```
$ iotop --help
```

Using `-o` option can be useful to avoid clutter.

```
student@ubuntu: ~
student@ubuntu:~$ iotop --help
Usage: /usr/sbin/iotop [OPTIONS]

DISK READ and DISK WRITE are the block I/O bandwidth used during the sampling
period. SWAPIN and IO are the percentages of time the thread spent respectively
while swapping in and waiting on I/O more generally. PRIO is the I/O priority at
which the thread is running (set using the ionice command).

Controls: left and right arrows to change the sorting column, r to invert the
sorting order, o to toggle the --only option, p to toggle the --processes
option, a to toggle the --accumulated option, i to change I/O priority, q to
quit, any other key to force a refresh.

Options:
  --version          show program's version number and exit
  -h, --help        show this help message and exit
  -o, --only        only show processes or threads actually doing I/O
  -b, --batch       non-interactive mode
  -n NUM, --iter=NUM number of iterations before ending [infinite]
  -d SEC, --delay=SEC delay between iterations [1 second]
  -p PID, --pid=PID processes/threads to monitor [all]
  -u USER, --user=USER users to monitor [all]
  -P, --processes  only show processes, not all threads
  -a, --accumulated show accumulated I/O instead of bandwidth
  -k, --kilobytes  use kilobytes instead of a human friendly unit
  -t, --time       add a timestamp on each line (implies --batch)
  -q, --quiet      suppress some lines of header (implies --batch)
```

14.9 Using ionice to Set I/O Priorities

ionice utility sets both I/O **scheduling class** and **priority** for given process. Takes the form:

```
$ ionice [-c class] [-n priority] [-p pid] [COMMAND [ARGS] ]
```

If **pid** given with **-p** argument results apply to requested process, otherwise it is process that will be started by **COMMAND** with possible arguments. If no arguments given, **ionice** returns scheduling class and priority of current shell process:

```
$ ionice
idle: prio 7
```

-c parameter specifies I/O scheduling class, which can have following 3 values:

I/O Scheduling Class

I/O Scheduling Class	-c value	Meaning
None or Unknown	0	Default value
Real Time	1	Get first access to disk, can starve other processes. Priority defines how big a time slice each process gets
Best Effort	2	All programs serviced in round-robin fashion, according to priority settings. The Default

Idle	3	No access to disk I/O unless no other program has asked for it for a defined period
------	---	---

Best Effort and **Real Time** classes take `-n` argument which gives **priority**, which can range from 0 to 7, with 0 being highest priority:

```
$ ionice -c 2 -n 3 -p 30078
```

Note: **ionice** works only when using **CFQ** I/O Scheduler (will talk about in next chapter).

##

[Back to top](#)

[Previous Chapter](#) - [Table of Contents](#) - [Next Chapter](#)