The sys admin's daily grind: Dstat

More Stats

Occasional worries about the system status are part of the sys admin's daily life, and admins usually keep a fat toolbox of *top* and *stat* tools to alleviate them. Charly says he can manage with just one multitool – for the time being, at least. *By Charly Kühnast*

How many times have I written about tools that have "top" or "stat" in their names? It feels like *umpteen* times. Today, Dstat [1] makes it *umpteen + 1*. This tool, which many distributions already have in their repositories, claims it will save me the trouble of making that *umpteen + 2* times, while sending many system reporting tools into a well-earned retirement [2].

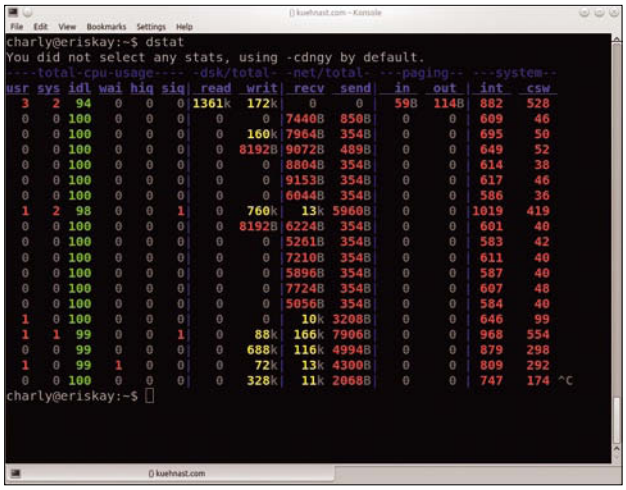

Figure 1: When launched without parameters, Dstat writes a new line with metrics every second.

Although Dstat suffers from no shortage of parameters, I first called it without any. The results are lines of the most important system data written once a second (Figure 1). But, if you want to drill into the depths of the network, you can use the `-tcp` and `-udp` switches. Using `-N eth1` restricts the output to a single interface. As for the network, other parameters can help bring details to light for all other system components.

That's all well and good, but the real fun with Dstat starts when I look at the list of plugins. The tool's entourage includes about three dozen of them, including a "Hello World" and a "Test" plugin that can serve as jumping-off points for your own add-ons. The plugins cover a considerable range of tasks. No fewer than five of them relate to MySQL, three cover InnoDB, and four cover NFS. Others show the Sendmail or Postfix queue length – even sorted by *incoming*, *active*, and *deferred* – and much more.

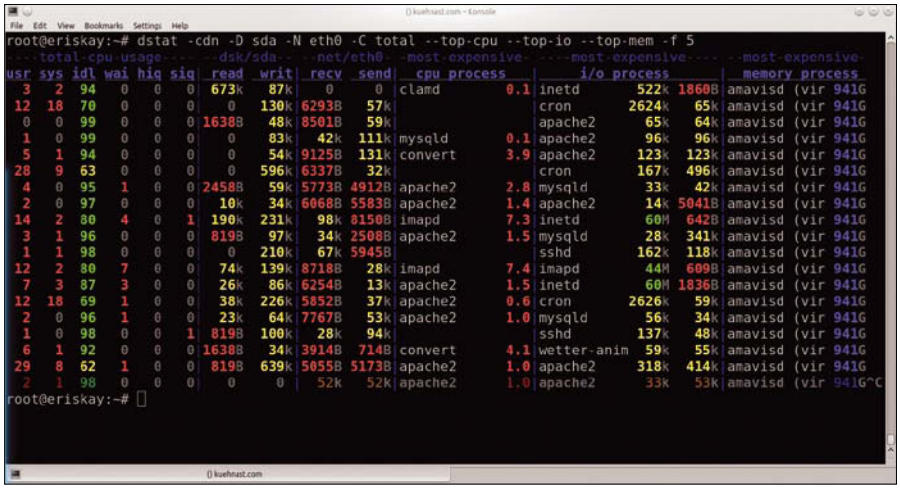

Figure 2: A parameter monster, admittedly, but this view of an overload scenario quickly puts you on the right track.

Top Picks

My personal favorites are the `top` plugins. They show you processes that stick out in a certain way. For example, `dstat --top-mem` points a finger at the process currently hogging the most RAM. On my desktop this is, not surprisingly, Firefox. The `--top-io` plugin reveals the process that is currently generating the most disk load and that is responsible for I/O operations subjectively feeling sluggish.

Dstat also lets me visualize coherencies in a great way. Besides CPU, disk, and network load, Figure 2 shows the processes that cause the most CPU, I/O, and memory load on a small server. This view is very handy because, if an overload occurs, it quite reliably points to the spot where the blaze broke out. Unfortunately, you do need a string of parameters from hell to conjure up this view:

```
dstat -cdn -D sda -N eth0 ➥
  -C total --top-cpu --top-io ➥
  --top-mem -f 5
```

If you want to visualize the values graphically, you will be pleased to hear about the possibility of writing all the output to a comma-separated file. However, only time will tell whether Dstat really does keep me from writing about `top` or `stat` tools *umpteen + 3* times at some point in the future. ■■■

INFO

[1] Dstat: *http://dag.wieers.com/home-made/dstat*

[2] "Stat Pack" by Valentin Höbel, *Linux Magazine*, June 2012: *http://www.linux-magazine.com/Issues/2012/139/System-Diagnosis-Tools/*