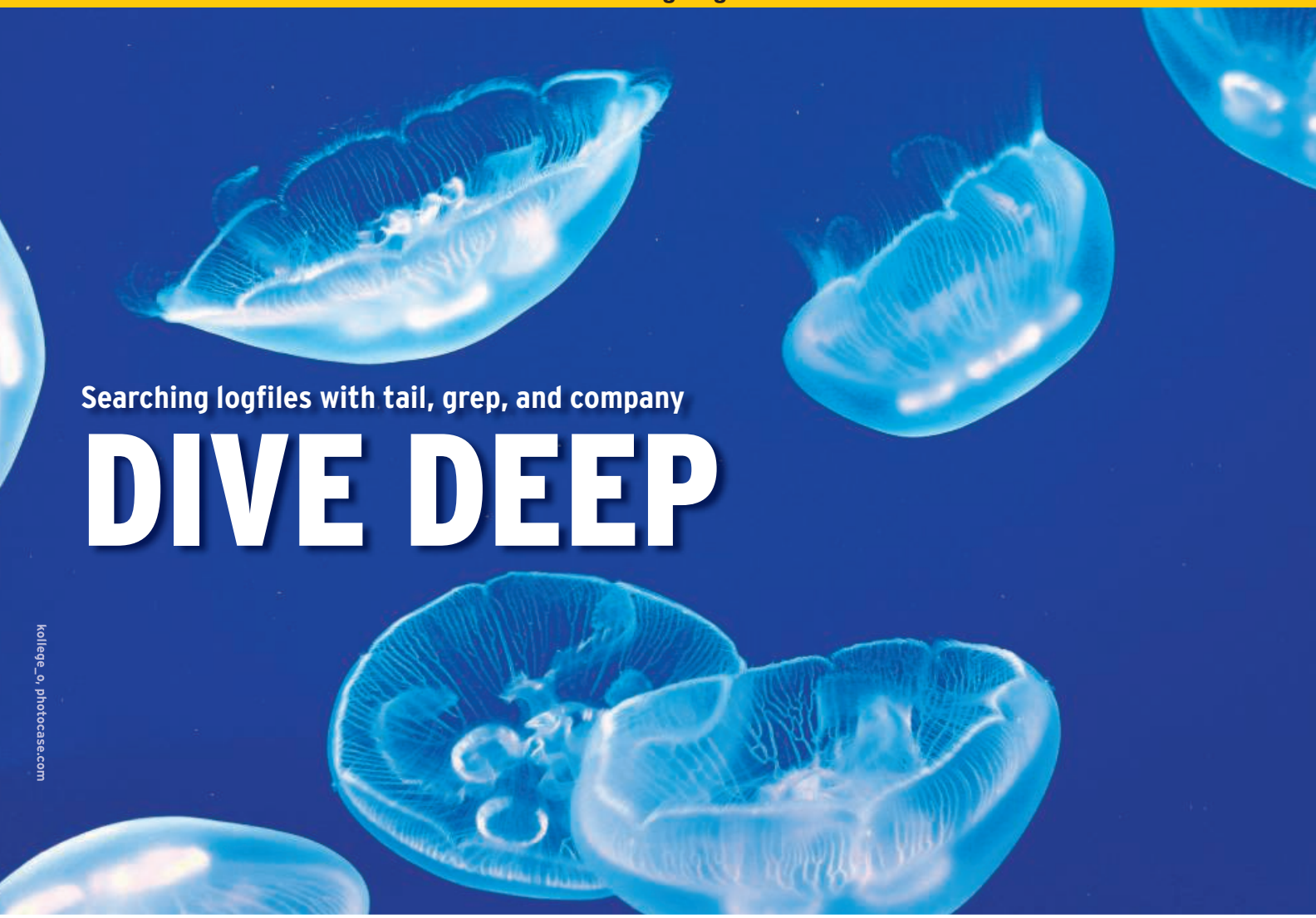

Searching logfiles with tail, grep, and company

DIVE DEEP

kollege_o, photocase.com

If your hardware or software goes on strike, or the graphical system or your Internet connection fail, checking the logfiles can often be a big help. In this month's column, we will look at the command-line tools that can help you scour the depths of these critical files. **BY HEIKE JURZIK**

Kernel messages, user logins or log offs, network processes, and many other events are logged meticulously by the Linux system. The Linux system's logging system goes by the name of *syslogd* (or *syslog-ng*, "Syslog New Generation" on SUSE Linux); the system logger is a daemon, which is started at system boot time. All log files are stored in the folder */var/log/* and its subdirectories (Figure 1).

The View

With just a couple of exceptions, most of these protocol files are protected from prying eyes and only readable by the system administrator. To view the files, you can use KDE's file manager, Konqueror, for example, in system administration mode. To do so, pop up a quick starter by pressing Alt + F2 and type *kdesu konqueror*, then type the root password after the prompt.

Because logfiles are text-only, you can view the content with any text editor. Of course, doing so is fairly tedious, and finding the information you need can take a while.

The following sections describe alternative command-line approaches and give you some troubleshooting tips.

Well Sorted

One of the most important logfiles – and the first place to look if something goes wrong – is */var/log/messages*. In this file, most distributions write messages about network connections, starting and terminating services, hardware drivers, user authentication, and more.

In contrast to this, most systems write information on the print system to the */var/log/cups/* folder. The logfiles below this folder may contain error messages, access to configured devices, and more. If the screen stays blank, the mouse fails to work, or 3D acceleration is not supported correctly, you will need to check out the */var/log/Xorg.0.log* file. Forums and mailing lists will be helpful if you can give references to the appropriate logfile sections.

Access Permitted?

If somebody tries to escalate their privileges to administrator level by entering *su* in a terminal window or by launching a distribution-specific setup tool, information on the attempt will be written to */var/log/messages* (or */var/log/auth* on some systems).

In addition to the date and time, the entry tells you which user initiated the command and whether it was successful. Whereas SUSE Linux only gives you details of the success or failure of the attempt, Mandriva Linux also tells you which program tried to gain root privileges. Listing 1 shows some important

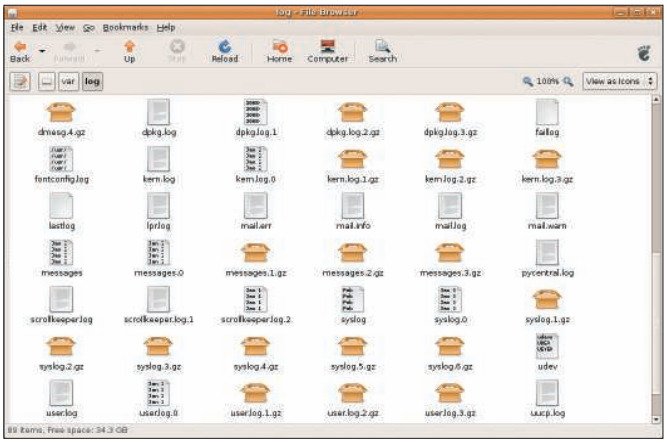

Figure 1: Log files are located below "/var/log/".

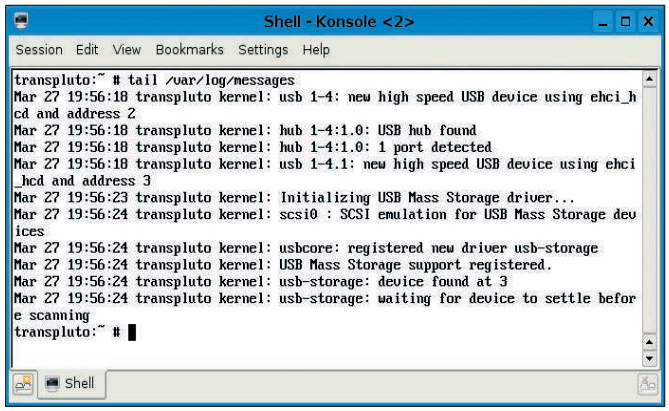

Figure 2: "/var/log/messages" shows you events such as plugging and unplugging media (a USB stick, in this case).

messages on a SUSE or Mandriva Linux system.

If your computer is assigned an IP address by a DHCP server, identifying DHCP client activity is just as easy. Lines like this

```
Mar 27 19:18:45 localhost ➥
dhclient: DHCPREQUEST on eth0 ➥
to 255.255.255.255 port 67
Mar 27 19:18:45 localhost ➥
dhclient: DHCPACK from ➥
192.168.2.15
Mar 27 19:18:46 localhost ➥
dhclient: bound to ➥
192.168.2.237 ➥
-- renewal in 235 seconds.
```

show you how your computer requests an IP address and details about its validity period.

If you use a direct Internet connection via modem, ISDN, or DSL, */var/log/messages* also will tell you whether your connection is working because the dial-up program *pppd* (modem and DSL) or *ipppd* (ISDN) writes its status message here; for example:

```
Mar 25 22:14:19 asteroid ➥
pppd[1432]: local IP address ➥
195.14.222.177
```

The Whole Truth?

If you are worried about a recent event, it makes sense to check the last few lines of a logfile. Instead of opening the whole file in your text editor and scrolling down to the bottom, you can use the *tail* pager at the command line.

Open a command line – for example, by typing *konsole* in the quick-start window that you popped up by pressing Alt + F2 – and then become root by typing *su* and entering the administrative password. Now call *tail*, passing in the logfile name, to show the last 10 lines of the file (Figure 2). If you need more than 10 lines, you can set the *-n* option to specify a different number, such as:

```
tail -n 20 /var/log/messages
```

If you see an error message like

```
Mar 27 15:43:21 transpluto ➥
kernel: usb.c: USB device 10 ➥
(vend/prod0x82d/0x200) is ➥
not claimed by any ➥
active driver.
Mar 27 15:43:25 transpluto ➥
/etc/hotplug/usb.agent: ... ➥
no modules for USB product ➥
82d/200/100
```

you can assume that the device has not been detected and will not be supported.

The *tail* program includes another practical feature: You can set the *-f* option to switch to infinite mode, where *tail* will update the file display whenever a file changes.

If you want to keep an eye on */var/log/messages*, type:

```
tail -f /var/log/messages
```

Then you can monitor the ongoing activities. Pressing Ctrl + C quits the display.

"tail" and "grep"

Finally, to search the output from *tail* for keywords, you can use it in combination with another command-line tool and discover critical messages far more quickly. The *grep* tool searches for patterns in strings.

If you want to search the last 100 lines of the */var/log/messages* logfile for "USB" or "usb", you can do so with a single command:

```
tail -n 100 /var/log/messages ➥
| grep -i usb
```

This command pipes ("|") the output from *tail* to the *grep* command; the *-i* parameter switches off case sensitivity (i.e., it does not distinguish between "usb" and "USB" or even "usB"). ■

Listing 1: Root Privileges Denied

```
01 # Unsuccessful attempt by user suse93 to gain root privileges on a
02 #Suse Linux System
03 Mar 27 14:30:51 transpluto su: FAILED SU (to root) suse93 on /dev/
   pts/6
04
05 #Mandriva control center launch with ensuing incorrect entry of
06 #the root password:
07 Mar 27 18:11:41 localhost drakconf.real[4395]: ### Program is
   starting ###
08 Mar 27 18:11:47 localhost su(pam_unix)[4404]: authentication
   failure; logname= uid=500 euid=0 tty= ruser=mandriva2006 rhost=
   user=root
```