Roots and boots

Chris Bidmead has a fix for those flummoxed by the tomsrtbt floppy.

I've had quite a bit of feedback on tomsrtbt, the Linux-on-a-floppy to which I introduced you a couple of issues ago (the home page has since moved to www.toms.net/rb). From your emails I see that many of you have had a problem creating the floppy. I told the first readers who emailed me with this complaint that their floppy drives might be too old and probably weren't capable of handling the 1,722Kb formatting that tomsrtbt requires. Sorry — this was probably rubbish.

What I hadn't realised was that many of you have been trying to create the disk from a Windows 95/98 DOS emulation screen. This won't work: the emulator doesn't connect to the diskette at a sufficiently low hardware level for the superformat utility to work. If you're a Windows user you'll need to reboot the machine into a real DOS session. But if you still can't cut it, there's an alternative small Linux that boots from a pair of 1,440Mb diskettes which you can download from www.trinux.org/. Alternatively, there's a single-floppy Linux implementation at http://206.47.27.32/~ebenoit/loaf/.

I'm getting to like tomsrtbt a lot. No, it doesn't give you a delicious GUI, but then, delicious GUIs can be pernicious to the extent that they hide away details about which you really ought to know. Boot up the tomsrtbt diskette and you find yourself at the command line with a limited subset of Unix tools with which to work. It's back to basics, but there's a lot you can do and learn with it. For example, if you've ever installed a Linux distribution — or a version of Windows, come to that — on a networked machine, you'll remember how the installation walks you through the process of setting up the network connection, asking you about network cards, dotted quad addresses, nameservers and suchlike. What exactly does it do with all this info? And later, when the net connection mysteriously falls over, what do *you* do?

Helpful GUI dialogue boxes often dangerously conceal the specifics of what's going on under the surface. They don't show you what you'll need to fix when things go wrong — and often it's not at all complicated, if you know what you're doing, which you don't because the GUI has blindfolded you.

Here's how I set up a network connection on my ten-year-old Nokia AlfaSkop 386 machine, using tomsrtbt. Insert floppy and boot. Have a cup of coffee — with the 16MHz 386, this part takes forever. However, once tomsrtbt is installed, it's the fastest way of running UN*X on the hardware because everything is carried out from RAM disks. As a corollary, another benefit is that whatever you already have installed on the machine doesn't get touched and cannot be harmed.

OK, Linux version 2.0.35 is now up and running. There is no network connection. I can see this by typing ifconfig at the prompt — it just gives me something like that shown in Fig 1. This means the

IS IT LINUX OR IS IT WINDOWS?

▲READER JON WILLIAMS AT JONW@NOUVA.COM HAS FOLLOWED UP ON THE KDE AND GNOME SCREENSHOTS (PCW, NOV '98) WITH A SUGGESTION THAT WE SPIN OFF AN ADDITIONAL KDE/GNOME COLUMN HERE. IF YOU AGREE, EMAIL US AT UNIX@PCW.CO.UK. JON HAS SENT ME THIS SCREENSHOT FROM HIS NOTEBOOK COMPUTER AS AN EXAMPLE OF HOW TO MAKE LINUX LOOK LIKE WINDOWS 95 'FOR THAT COMFORTABLE FEELING'... HUH! DEBATABLE.

[FIG 1] The missing link

```
lo    Link encap:Local Loopback
      inet addr:127.0.0.1  Bcast:127.255.255.255  Mask:255.0.0.0
      UP BROADCAST LOOPBACK RUNNING  MTU:3584  Metric:1
      RX packets:1 errors:0 dropped:0 overruns:0
      TX packets:1 errors:0 dropped:0 overruns:0
```

loopback connection internal to this machine is established, but there's no link to the outside world. I can fix this with

```
ifconfig eth0 192.168.1.1
```

or, in plain English, "Configure the interface called eth0 as 192.168.1.1".

Damn! I get an error message: "Unknown interface". I have tried to give eth0, the ethernet device, a dotted quad address but there's no such thing. Well, there is *physically*; it's a very old 8-bit NE1000 but the software doesn't know about it.

With the latest versions of Linux, this NIC is driven by a dynamically loadable module, ne.o. Normally I would run lsmod, the list modules utility, to see if this module is loaded, but tomsrtbt misses out lsmod to economise on space. But you don't really need it. Linux uses a directory called /proc, a kind of fake filesystem where you can peer into the heart of your Linux installation by looking in the appropriate files or subdirectories. In this case, /proc/modules tells the story — there's no ne.o loaded.

OK, so I insert the module with insmod ne. There are no messages in response to this, so it looks as if this has succeeded (Unix conventionally respects terseness and only speaks up when things go awry). Inspecting /proc/modules confirms that ne.o is now running. Now I can repeat the ifconfig exercise above — successfully this time.

I still can't ping anywhere yet (except through my own loopback) because I haven't yet set up a route between my now working eth0 and the network itself. The command which does this is called, reasonably enough, "route". It works like this:

```
route add -net 192.168.1.0 → eth0
```

which means "add eth0 to the existing network whose name is 192.168.1.0". The 0 at the end means the whole network.

Another TCP/IP convention is that 255 at the end means "every device on the network". Pinging 192.168.1.255 is a useful way of seeing what's running. The result is a working network connection.

This is relatively easy for me because I've done it before, but even if you're a complete beginner you should not be at a loss with tomsrtbt because, despite its compact size, there are manual pages for most of the utilities it provides, including ifconfig and route. "But a complete beginner will not even know to look for the manual pages of ifconfig," I hear you cry. Well, you're right. But I am visualising a proactive beginner who is inquisitive enough to have stumbled on the documentation in /usr/doc — the standard Linux place to keep documentation — and has followed the trail from there. Tomsrtbt is not much use for the uninquisitive, but then, I didn't say it was.

You will be able to find an alternative approach to Linux-on-a-high-density-floppy, called MuLinux, at http://www4.pisoft.it/~andreoli/mulinux.html. I have not yet tried it, so if you get there before me, do please let me know what you think of it.

THE COBALT CUBE

Last month, separately from this column, I reviewed the Cobalt Qube 2700, a neat hardware add-on which brings web serving and free text retrieval to any network. It happens to run Linux, so it's easy to hack it to do a lot of other things. John Adams at the UK supplier, Mintra (phone 0161 256 4030), is very kindly allowing me to experiment with it for a while. Initially, I plugged it into my own network here as yet another regular Linux machine, with the interesting twist that it's based on a Mips processor rather than an Intel chip. I may yet get around to that, but it's turned out that the bundled software — a web server, a document organiser and some web-based system management tools — really do turn this attractive little blue box into a compelling, self-sufficient "Intranet Device". So far, I've found a lot of work for it in that mode. Check out the screenshot [Fig 2].

◀ FIG 2 HERE'S PART OF THE COBALT QUBE WEB-BASED SYSTEM MANAGEMENT. AN AUTHORISED SYSADMIN CAN MANAGE THE QUBE FROM ANY WORKSTATION WITH A WEB BROWSER

Intel to the rescue

A word of thanks is due. Five years ago, when I originally set up the multiplatform network around which this column is based, I installed a six-port 10BaseT TigerHub from SMC to connect my newer 10BaseT networked machines to my legacy 10Base2 "Cheapernet" coaxial network. For half a decade, that hub has been a faithful ally, although running hot enough to fry an egg on.

A good network component is an invisible component, and frankly I've hardly given it a thought since it was first installed. Until last week...

I sat down at my desk in the morning, only to discover that the network was down. The TigerHub had finally burned itself out. I called Steve Roberts, who

handles networking at the UK Intel press office, and he rushed over one of Intel's sleekly designed 10BaseT InBusiness hubs: an eight-port hub this time, which gives me room for some expansion. Hub design seems to have improved in the interim because this one runs cool, so it should be around for a long time. Thanks, Steve.

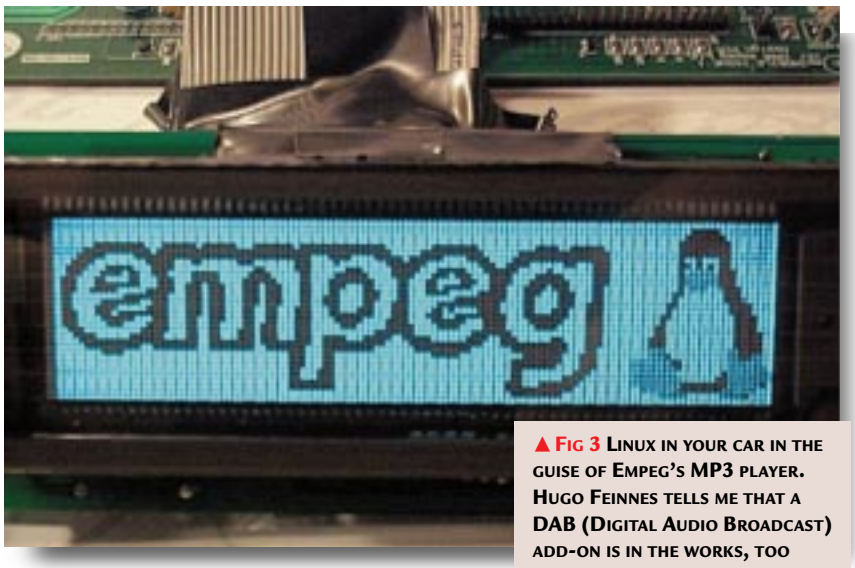
Motoring news

I don't drive a car, which perhaps is why I found a recent Microsoft demo of its proposed Windows-CE-in-your-dashboard so hilarious. Codenamed Apollo, the Auto PC is an "in-car multimedia system" which, among other things, reads your emails aloud to you as you drive and no doubt lets you surf the Superhighway by voice control while you're stuck in traffic jams.

Back in the real world, a small UK outfit called Empeg has come up with an eminently practical way of employing computer power behind the wheel. The Empeg [Fig 3] is a dashboard device which delivers CD-quality music you've previously compressed to the MP3 standard on your PC, and then loaded onto the Empeg's internal 2.5in hard drive. The base model comes with a 2.1Gb hard drive that gives you 35 hours of non-stop playing time. Not enough for you? Well, you can fit 16Gb-worth of hard-drive capacity (more in future) into the 6x5x2in unit and drive around with the equivalent of a 300-CD collection which will play non-stop for almost a fortnight!

At the time of writing the price had not yet been fixed for the unit, but it's expected to be about £600 (ex VAT). This includes a 2.1Gb drive and an RDS FM tuner, as well as a docking station, cables (USB and serial), PC software, extra power supply unit, manuals and an installation guide.

Very nice, but what's this got to do with Unix? Well, because the thing's really a computer, it needs an operating system. No prizes for guessing which one Empeg is using. The animated penguin on the display should give you a clue. The bad(ish) news is that the software Empeg provides to compress and download the music only runs under Windows. Empeg plans to make some (non-GUI) Linux software available to do this and the company will be publishing the specification of the transfer format for the unit. Read more about it at www.empeg.com.



▲ FIG 3 LINUX IN YOUR CAR IN THE GUISE OF EMPEG'S MP3 PLAYER. HUGO FEINNES TELLS ME THAT A DAB (DIGITAL AUDIO BROADCAST) ADD-ON IS IN THE WORKS, TOO

If you want to save a few bob and build your own machine, Hugo Feinnes, the engineer working on the project, has very generously laid out the full details of his prototype, the mp3mobile, at http://utter.chaos.org.uk/~altman/mp3mobile/.

Beginning again

Dougie Richardson <dougie@shadow.abel.co.uk> writes to tell me that Linux UK <www.linuxuk.co.uk> has started a new section for Linux beginners. It comes in two parts: a monthly guide to particular topics, like dialup networking, and a beginner's Q&A session. Dougie is inviting readers to contribute questions to the Q&A, as well as ideas about what they'd like to see in the step-by-step guides. In a sense, I suppose, Dougie's venture is a rival to this column. But in the free-software community, we welcome alternative sources and diversity.

One caveat for Dougie: he needs to stand by for flack...

One caveat for Dougie: he needs to stand by for flack. I can tell him from experience that nobody gets to pronounce on Unix in public without receiving a great deal of incoming, er, corrective punishment. For example, his web page as it stands at the time I'm writing this asserts that Unix filenames cannot contain spaces. It's an easy mistake to make if you don't know that you can always wrap a filename in quotemarks:

```
touch "Spaces are just fine"
```

creates an empty file of that name with no problems on any Unix system I've used. And when Dougie claims "In theory, the filenames you use in Linux can be up to 256 characters, although in practice most of us keep it down to 14", I don't know who he means by "most of us".

Speaking for myself (which is about as much as I dare in this column) my filenames will conform to the traditional Unix penchant for terseness where appropriate (why call a Perl test program "testing.prl" when you can call it just "t"?). Alternatively, if I'm about to mess with an XF86Config file, copying the old one to XF86Config.orig is unimaginative. I am just as likely to give the copy an informative name such as "the XF86Config that works alright with my Millennium II but flickers a bit at high resolution". And how about when I backup my PalmPilot to a directory called "PalmPilot dump as of 'date'"?

But Dougie's big advantage is that as the flack flies in, he can fix his web page. My goofs in print, and I've made plenty, stay on the record.

PCW CONTACTS

Chris Bidmead *can be contacted via the PCW editorial office (address, p10) or email* **unix@pcw.co.uk**.