A Linux Adventure

By Rob Bracken

Planning the Route

I recently needed a Linux PC for a talk I was due to give to a BUG meeting ("ABC of X" – if you were there, I hope you enjoyed it). I have Linux installed on a mini-tower PC, but it was too bulky and heavy to take on the train so I would have had to load it into the car and drive to Hammersmith. I also have a laptop, whose 2GB disk is stuffed full with Win95 and other client-related stuff. Easy! – just backup the whole laptop disk, overwrite it with Linux and restore it when I'd finished. Not so easy – the laptop has no CD-ROM drive!

Of course, the easiest option would be to resign myself to a long drive, but being the sort of person who relishes the challenge of a difficult problem, I decided to have a go at my laptop.

Packing the Bags

The first part was easy. The laptop has a network card, so I used Win95 Backup to save the whole disk in a file on my server. I also made a Startup disk and wrote down the details of the disk partitioning (one big *FAT* partition). I ran into trouble, however, when I tried to work out how I would restore Win95 when I was finished.

As I mentioned, the laptop has no CD-ROM drive, so I wouldn't be able to reload Win95 from CD. I could boot Win95 from the Startup disk, but it had no network support, so I couldn't restore from the backup file on the server. At this point, I started to think about using Linux.

Catching the Bus

Linux has networking built in at a low level, so I reckoned I'd be able to boot my laptop from a Linux boot disk and connect to the other machines on the network. The SuSE manual has a couple of paragraphs about network installation of Linux. It isn't very clear or encouraging – but worth a try, I thought.

The SuSE manual also talks about a boot disk and a rescue disk. You can boot Linux from the boot disk and then load various utilities from the rescue disk. It's designed to help you recover from disasters, and it looked like it was exactly what I needed to make a Linux backup of my FAT partition. (This may also be called an *emergency disk* in other distributions).

The Train Journey

The rescue disk contains the compressed image of a small, root filesystem. SuSE supply it in the file */disks/rescue* on the 1st CD-ROM. It's possible to use the file directly across the network, but I chose to keep things simple and copied it to a floppy (Mount the CD-ROM on e.g. directory */cdrom* and copy the file with *dd if=/cdrom/disks/rescue of=/dev/fd0 bs=18k*).

Next problem – I wanted to make a Linux backup of the laptop FAT partition onto another Linux machine, across the network. How could I connect to the other machine? I booted SuSE and had a look.

At the Airport

The SuSE boot disk asks a few questions about which country you're in, whether or not you've got a colour display and which keyboard layout you're using. It then goes to a *Main Menu* with these options:

Settings
System information
Kernel modules (hardware drivers)
Start installation / system
End / Reboot.

The network card in my laptop is a *PCMCIA* card, and there was a *PCMCIA* option in the Kernel modules, so I tried it. SuSE asked for the modules disk and then said it had found a *PCMCIA* chipset. It asked for parameters, which I left blank (mainly because I didn't have a clue what they might be!). After a few bleeps, it displayed a success message, followed by some Kernel messages, including a reference to *eth0* – a network interface. "Aha!" I thought, "We're in business!".

The next step was to go back to the *Main menu* and select *Start installation / system*. This took me to another menu with these options:

Start installation / update
Boot installed system
Start rescue system
Start Live CD
Eject CD.

I put the Rescue floppy in and selected *Start rescue system*. A menu appeared asking for the source media – I selected *Floppy*. The filesystem loaded and Linux booted! A quick look at */bin* showed a number of Linux utilities available, including *mount*, *umount*, *gzip*, *gunzip* and *tar*.

In-flight Refreshment

Next problem – how do I start the network card and connect to a directory on my Linux PC? Starting the network card turned out to be easy. The *ifconfig* program handles this sort of thing, so *ifconfig eth0 10.0.0.200 up* started the network card (*eth0*) and gave it an IP address of *10.0.0.200*. I confirmed this by typing *ifconfig* with no arguments. (If you want to stop the network card, type *ifconfig eth0 down*.).

I could now *ping* the Linux PC, but I couldn't save a backup to it yet. I had to create a directory on the Linux PC – which I called */var/backups* - and *export* it. (Add the following line to */etc/exports*: */var/backups (rw,insecure)*. Execute *rcnfsserver restart* to load the new settings into the nfs server.) Then create a */var/backups* directory on the laptop and *mount* it across the network. (*mount –t nfs 10.0.0.100:/var/backups /var/backups* where *10.0.0.100* is the IP address of the PC that's *export*ing the directory and */var/backups* is the directory you want to *mount* it on.)

On Safari

I had finally got to the point where I could save a Linux backup of my Win95 hard disk. I created a */mnt/cdrive* directory and *mount*ed the hard drive on it. (*mount –t -r vfat /dev/hda1 /mnt/cdrive* – the *–r* option mounts the drive read-only. Use *vfat* to handle long filenames correctly.) I saved a *tar* archive file in the */var/backups* directory (which was really the directory on the Linux PC). (*tar cvzf laptop.tar.gz /mnt/cdrive* creates a *laptop.tar.gz* archive file of all the files in */mnt/cdrive* – the *z* option zips the archive as it's written.)

So, I had saved a *tar* archive of the laptop's hard drive, without needing to install Linux on the laptop. I unmounted the */mnt/cdrive* and */var/backups* directories (*umount /mnt/cdrive, umount /var/backups*), put the SuSE boot disk back in and rebooted the laptop (*shutdown –r now*). Again, I loaded the *PCMCIA* module (to load support for the network card). I exported the CD-ROM drive from the Linux PC (add */cdrom (ro,insecure)* to */etc/exports*, load the new settings with *rcnfsserver restart*, mount the CD-ROM drive with *mount /dev/cdrom /cdrom*) and selected *Start installation / system*. When the installation program asked me for the source media, I selected *Network (NFS)*. I answered *No* to *Automatic configuration via bootp*, and entered the following settings:

IP Address	*10.0.0.200* (arbitrary value – can be anything that doesn't conflict with other addresses on the network)
Netmask	*255.255.255.0*
IP address of gateway	*10.0.0.200*
IP address of name server	*<ESC>*
IP address of the NFS server	*10.0.0.100* (the address of the Linux PC whose CD-ROM drive I want to use)
Directory on the server	/cdrom

At this point, the standard SuSE installation started and I was able to wipe my hard disk and load Linux.

The Return Journey

After the talk, I had to restore the laptop to its Windows 95 state. I booted it from the boot floppy and loaded the rescue system. I used Linux *fdisk* to wipe the Linux partitions. Boot the Windows Startup disk, use *fdisk* to create a single *FAT* partition and format it. Reboot the Linux rescue system. (It would have been easier to use Windows 95 *fdisk* throughout, but it didn't recognise the logical partition that I'd created when I installed Linux). I then started the network card, mounted the */var/backups* directory from the Linux PC and mounted the hard drive. "Great" I thought, "Nearly there" – but not quite.

I started restoring the hard disk from the *tar* archive, but I noticed error messages saying that files couldn't be found. I stopped the restore and checked the hard disk. It looked OK, but I noticed that directories like *Program Files* were missing. I tried to make a *Program Files* directory and got an error. I tried it on the Linux PC – it worked OK. The problem was that the version of *mkdir* on the rescue disk was earlier than the version that's now in use, and it doesn't support spaces in directory names! (I expect SuSE did this to save space, as the earlier library is smaller).

My solution to this was to use the Linux PC's versions of the programs. This involved exporting the root filesystem of the Linux PC (add */ (ro,insecure)* to */etc/exports* and execute *rcnfsserver restart* to load new settings. You have to remove or comment out other entries, as you're not allowed to export directories twice, and any other directory will be exported implicitly by exporting the root), mounting it on the laptop's filesystem (make a */mnt/PCroot* directory, mount it with *mount –t nfs 10.0.0.100:/ /mnt/PCroot*), and putting its */bin* and */sbin* directories at the front of the *PATH* variable (*export PATH=/mnt/PCroot/bin,/mnt/PCroot/sbin,$PATH*). I also added symbolic links to the */lib* directory pointing back to the Linux PC's */lib* directory, so that the later versions of the programs could find their libraries when I ran them on the laptop (*ln –s /mnt/PCroot/lib/library /lib/library*). I made links to *linux.so.2, libc.so.6, libdl.so.2, libm.so.6, libncurses.so.4* and *libncurses.so.5*. I could now make a *Program Files* directory on the laptop.

I started the restore from the *tar* file again – this time there were no error messages. I rebooted the laptop – it couldn't find the operating system! Fortunately, this was easily fixed by booting the Windows 95 Startup disk and executing *fdisk /mbr* to restore the *master boot record* to the hard drive. Now the laptop booted OK. Everything looked as it was before. Just to make sure I verified the disk against the Windows backup I'd saved. The only differences reported were in log files, which I would expect to see as they are updated in normal use.

Home to Bed

So I'd achieved my goal of saving a backup of my Windows 95 laptop, installing Linux without a CD-ROM drive and restoring Windows 95 – also without a CD-ROM drive. Along the way, I learnt a lot about using Linux across a network. Note that the *tar* archive contained all the Windows registry files, without needing to do anything special. I hope this account of my venture into (for me) uncharted territory encourages you to try networking Linux for yourselves.

Rob Bracken is Managing Director of Bracken Software Limited and has over 10 years' experience of trying to make different programs talk to each other. You can contact him on rob_bracken@bramway.freeserve.co.uk.