

Access more than one Linux system

MULTIBOOT

Colin Murphy explains how, without too much effort, you can give yourself access to a new Linux system to play around in

New versions of GNU/Linux are being made available all the time. Sometimes it's hard to keep up with what's going on. You'll hear about lots of exciting developments, which you might like to try out, or at the very least play with. The problem with the latest developments is that they lie on the cutting edge of technology, so you may not find them in stable systems.

Obviously, any sensible person will want to play it safe and only use a stable system they trust for their day-to-day work. So how can anyone adventurous enough manage to get the best of both worlds?

Dual booting is something that's very common amongst Linux users, it's the most useful way of keeping more than one operating system on just the one machine. Normally a dual boot system will have some flavour of Windows, be it 95 or XP, as well as something useful, like a Linux system. But there is no reason why you can't have more than one Linux system on the same machine, even if they are of different distributions. Sharing information between these systems is also pain free.

Methods of booting

If you are creating a dual boot machine for the first time you need to make sure that you take some precautions. Make sure you have a boot disk for your current install; you will be changing things in your standard boot loading system, so if something does go wrong you need to know that you can get your system back up. There is the command line instruction *mkbootdisk*, which also needs to know the version number of the kernel that you will boot, which you can find with *uname* doing something like this:

```
# uname -a
2.4.8-26mdk #1 Sun Sep 23 17:06:39 CEST 2001
i686 unknown
# mkbootdisk 2.4.8-26mdk
```

and the boot disk will be made.

It is also useful to know what partitions on the hard drive get mounted to. This information is held in the file */etc/fstab/* and it's always a good idea to keep an up to date printed copy somewhere safe with your boot floppy and rescue disk set.



Here are the partitions set aside for the new install. You only need a '/'. You don't need a separate '/boot' or 'swap space' – learn by my mistake

We are making the assumption here that you have another hard drive onto which you are going to put this installation. This need not be the case – you can use any free space you have on your existing drive – but using a separate drive gives you another degree of separation from your existing, stable system.

During this new install you will be asked about making partitions for the new system. You should have a partition set aside for the new '/' partition, and I like to have a separate */home* partition as well. You don't have to make a new */boot* partition.

You may also want to mount some of your original partitions on this new installation, this would give you access to your original */home* directory, for



Here you can see how I have mounted my original '/home' partition in the new installation, giving it the name '/mnt/home2'

instance. Because you have a print out of `/etc/fstab` from your original install you can see which partition needs to be mounted.

There are two ways in which you can boot into your new system. The first would be to rely on boot floppies, at least for booting any secondary installation you make. The advantage of this is that you don't need to touch the boot loader for your existing stable system, the system that you most want to keep intact and in working order. The only real disadvantage of using a boot floppy is that you end up with a system that isn't quite as streamlined – you've got to remember to put the floppy in if and when you want to boot your new system, and you have got to remember to take the floppy out should you want to boot to your original, stable system. These might not seem like great disadvantages, but once you've booted into the wrong system three or four times you soon hanker after something less cumbersome.

The second, less cumbersome way is to amend the boot loader you have already so that you are given the option of loading whichever Linux installation you want, in very much the same way that you can choose between a Windows or Linux boot. The disadvantage is that you are now playing with a component of your stable system. However, if you've made your boot disk as we suggested then you can still get access.

Should you choose the first option then, when you start to install your new system, you should select something like 'expert' mode, where you are given the option of making full choices along the way. The standard install options will automatically make a boot device on your hard drive, this is the case with Mandrake at least. When you get to the 'install bootloader' section, or equivalent for your distribution, you need to change the boot device to `/dev/fd0`.

Taking the second, more streamlined option of installing a new bootloader, you should again select 'expert' mode to give you as much control over the process as possible. This time, once you reach the 'install bootloader' section, you will need to leave the boot device set to your hard drive. You can then add



The addition entry in the boot loader configuration screen for our existing system.

the details of your existing system to this new boot loader. You might also want to set the default boot image to be that of your original install, if you think that is the one you will be booting into most often.

Wrapping things up

That should be about it for the new install.

Depending on which method you have chosen, you can now boot into your new system by making sure your new boot floppy is loaded or by selecting the new image from the boot loader.

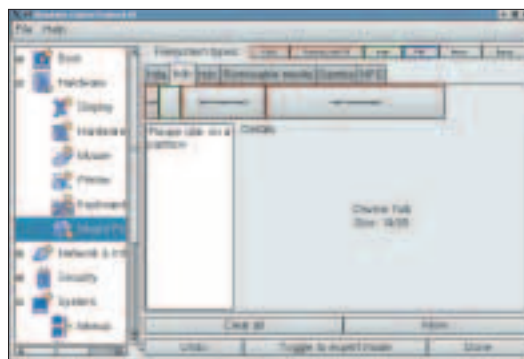
There is just one more thing to play with, you may want to give yourself access to your new `/home` directory from your original install. This is easy to set up, you need to add an entry to your original `/etc/fstab` file pointing to the new `/home` directory, for the sake of simplicity I mount it on `/newhome`.

Having access to the `/home` directories on your alternative installation is useful for passing data between the systems, but you should use caution when you try and share things like configuration files from utilities. The format of these configuration files might change between versions, and after all, you've gone through this process just to give you access to cutting edge versions, and you may be left with a configuration file that is incompatible with your existing application.

So you now have a new, secondary Linux installation with which you can thrash about in, installing and breaking things to your heart's content.



The 'Install bootloader' section in Mandrake. To stop your existing bootloader from being tampered with change the 'boot device' to `/dev/fd0`



Using the graphical configuration tool to add the newly created `/home` partition to my original install, calling it `'/newhome2'`. `'/newhome1'` is really the new `'/'` partition

But you should use caution when you try and share things like configuration files