



Klaus Knopper is the creator of Knoppix and co-founder of the LinuxTag expo. He currently works as a teacher, programmer, and consultant. If you have a configuration problem, or if you just want to learn more about how Linux works, send your questions to: klaus@linux-magazine.com

Password Type

? I have had my IBM ThinkPad R50e for about three years, and I have been running Ubuntu Linux on it reasonably satisfactorily. Originally, I used Ubuntu 6.06 LTS and could only get the wireless networking to work in open mode.

I now have Ubuntu 8.04 LTS installed and can use WPA2 with the password setting that my router requires. However, I have to set the network password type and password each time I switch the laptop back on. The password type has changed from WPA2 to WPA and the number of password blobs has increased from the correct 12 to more than will fit in the text box.

ASK KLAUS!



I attach the *lspci* details. The networking worked immediately, except for the annoyance of having to re-enter the password and type each session. Can you suggest anything I might do to make things better?



Please check that the password is correctly saved in `/etc/wpa_supplicant/wpa_supplicant.conf`, and ensure that *wpa_supplicant* has been automatically started once your computer boots.

The entry in *wpa_supplicant.conf* should be similar to Listing 1.

Some distributions set up *wpa_supplicant* via *wpa_cli* and entries in `/etc/network/interfaces` instead, but the basic configuration idea is the same. *wpa_supplicant* needs a while to authenticate, so maybe in your setup, the DHCP request just comes a little too early.

To check on the WPA status, enter the following:

```
$ sudo wpa_cli status
```

Listing 1: /etc/wpa_supplicant.conf

```
01 ctrl_interface=/var/run/wpa_
   supplicant
02 ap_scan=1
03 fast_reauth=1
04
05 network={
06     scan_ssid=1
07     # Use proto="WPA" to limit to
   WPA ONLY, "RSN" for WPA2
08     ssid="your_wlan_network_name"
09     psk="your_secret_wpa_password"
10 }
```

which should give an output like the following when WPA is operational and an IP address has been obtained:

```
Selected interface 'eth1'
bssid=00:01:02:03:1b:89
ssid=my_network
id=0
pairwise_cipher=CCMP
group_cipher=TKIP
key_mgmt=WPA2-PSK
wpa_state=COMPLETED
ip_address=10.0.0.100
```

Why Windows?



In the September 2008 issue, page 56, Klaus rhetorically asks why anyone would need Windows. I work for a company that uses VPN security to allow us to log in from home. This security package only runs on Windows XP (not earlier or later versions) and is called *SC_XP_2k_ngxR60_hfa1.exe*. Of course, this works in a virtual WinXP window, but this critical application must have Windows.

For those of us who cannot select the software packages that big organizations use, we are stuck with their choices. These choices almost always mandate Windows. In my case, it must be WinXP.



My rhetorical answer would be: "Why choose a proprietary VPN with an uncertain security level and uncertain future maintenance costs over a standardized, open source solution like OpenVPN or SSH?"

I'm not attempting to discuss whether or not *SC_XP_2k_ngxR60_hfa1.exe* would run under Wine, because in my opinion, it does not offer much advan-

tage using an open source operating system just as a loader for a proprietary software base.

Of course, you are right: Often the users are simply not allowed to make a choice on their own, which is a problem that, sadly, has no technical solution.

USB Modem



I am currently running Debian 4.0 “Etch” on a home-built PC (AMD Athlon, 1GB DDR RAM).

I’m trying to connect to the Internet using a USB modem (Huawei E160G) in the UK. I have managed to load the drivers for the modem using *dmsh* and *lsusb*, in conjunction with *modprobe* alterations. However, when it comes to configuring the modem using *kppp*, I need to provide usernames and passwords, which my ISP tends not to supply, therefore making it impossible to connect.

I decided to install Debian 5.0 “Lenny,” which I found to be a lot better when it comes to hardware detection. Lenny sees my modem as a GSM modem, without me making any alterations. However, when I try to connect, I am faced with similar unfriendliness regarding *kppp*.

I know that “Lenny” sees my modem but cannot get past this problem of configuring it so that it actually works. I have followed numerous online tutorials, only to be let down. Wvdial and Gnome PPP are equally unfriendly. My whole problem is probably between the chair and keyboard.

Also, will future Knoppix distros provide support for USB modems?



Debian/Knoppix, as well as all recent GNU/Linux distributions, support USB modems. A USB

serial module called USBSerial works with many devices that identify themselves as serial modems. Check with *lsmod | grep usbserial* for the presence of this module once the modem has been plugged in, and check *dmesg | grep ttyUSB*. If it doesn’t appear, try *modprobe usbserial* as root. For your modem, the serial line autodetection seems to work, so */dev/ttyUSB0* is operational.

The E160G is an UMTS modem, which means that the connection procedure is similar to dialing a provider’s number via phone line, except that some parts have to be changed in the standard setting of your modem program.

I’m describing two alternative configurations here: One is via *kppp*, the graphical modem tool in KDE, and the second one is a plaintext configuration of *wvdial*.

Don’t worry about the unknown login/password for authentication. Actually, you don’t need them because your modem identifies you using the SIM card number, so your Internet connection gets charged to your phone bill. But because *kppp* demands a login and password, you can literally give both of them as *three*.

Kppp Setup

Kppp’s default setting (probably?) has a bug. In the Debian branches I tested in the file */etc/ppp/peers/kppp-options*, the option *noauth* is commented out with a hash mark (#). For dialout, the option should be present, so please remove the hash mark or add *noauth* to *pppd*’s standard options in */etc/ppp/options*. This option for *pppd* just means that the address being connected to your modem is not required to authenticate itself, which makes sense here because you are not planning to let the modem log on to your computer, but you want to log on to the provider’s network instead.

After fixing that, you need to tell *kppp* that your modem is actually connecting via UMTS or GPRS, which is an init

Figure 1: Configuring modem settings with *kppp*.

Listing 2: */etc/wvdial.conf*

```
01 [Dialer Defaults]
02 Init1 = ATZ
03 Init2 = ATQ0 V1 E1 S0=0 &C1 &D2
    +FCLASS=0
04 Init3 = AT^SYSCFG=2,2,3FFFFFFF,1,2
05 Init5 =
    AT+CGDCONT=1,"IP","3internet"
06 Modem Type = Analog Modem
07 Baud = 460800
08 New PPPD = yes
09 Modem = /dev/ttyUSB0
10 ISDN = 0
11 Stupid Mode=1
12 Phone = *99#
13 Password = three
14 Username = three
15
16 [Dialer 3g]
17 Init4 = AT^SYSCFG=14,1,3FFFFFFF,2,4
18
19 [Dialer gprs]
20 Init4 = AT^SYSCFG=13,1,3FFFFFFF,2,4
```

string that has to be sent to the modem before dialing.

For your Internet connection, add

```
AT+CGDCONT=1,"IP","3internet"
```

as the Init2 string in the kppp Edit Modem Commands options (Figure 1), which differs among different providers.

For UMTS or GPRS, the number to dial is always (without the "ATD" prefix) *99# (including the asterisk and hash mark). Before your modem is ready to serve as a modem, you must unlock the SIM for this session. To do so, you either need to remove the PIN that protects the SIM Card from unauthorized use, or send the PIN once while the modem is plugged in. The modem command for this purpose can be typed directly in the kppp "terminal" that you can open in the modem command window,

```
AT+CPIN=1234
```

where 1234 is replaced by your SIM's pin number. If you mistype the number, the modem will greet you with a "PUK Pin" instead of "SIM Pin" request, which means you will have to unlock the SIM with the much longer PUK that was provided with your SIM Card. Failing authentication will result in the modem responding with something like "illegal command" to every dial attempt.

Now you are ready to connect. You can save all settings in kppp; only the PIN command above must be typed each time you remove and reconnect the modem from the computer's power, unless you disable the PIN on your SIM Card.

Wvdial

An anonymous poster on the Ubuntu forums at <http://ubuntuforums.org/showthread.php?p=5569939> reports that the `/etc/wvdial.conf` entry shown in Listing 2 worked for him.

Note the different settings for fast UMTS and the slower GPRS variant. Wvdial has to be started as root to fire up pppd after connecting.

Stick Script



In the October 2008 issue of *Linux Magazine*, you told of the script to put Knoppix 3.5.1 onto a USB stick.

I have a 4GB (3950Kb or something) USB stick, and it is not big enough. It bombs out with an error stating that there is not enough space. What size stick do I need? I was under the impression that this type of thing could be done on a 1GB stick.



The CD version of Knoppix is about 700MB, whereas the DVD version is over 4GB. The CD version should fit on a 1GB stick. The DVD version requires a stick of around 5GB, so the closest common size available is probably an 8GB stick for the DVD version (which I frequently use for testing USB boot compatibility).

Also, if you remaster the compressed files, please make sure the resulting size of a single file after compression does not exceed 4GB, because that is the maximum file size supported by the FAT32 filesystem, which is usually used for USB sticks.

The same limit applies for a plain ISO 9660 CD-ROM/DVD filesystem. Thus, to stay below this limit, the compressed files are split into two parts on the Knoppix DVD, then they are merged via UnionFS/AuFS during the boot process.

SATA Hard Drive



My desktop now has only a SATA hard drive. It seems that your UnionFS cannot be formulated. In principle, your Live DVD system does not use any hard drive, but do you use a hard drive cache somewhere?

Knoppix 4.x.x up to 5.3.1 all behaved the same.



In Knoppix, `/UNIONFS` is the directory in which the read-only, compressed data meets the writable ramdisk, and (almost) all system directories are linked to this directory during the boot procedure.

This arrangement is entirely independent of any hardware, except maybe the CD- or DVD-ROM, which is expected to be mounted at `/cdrom`. Knoppix does not use the hard disk by default.

However, it is still possible to put a *knoppix.img* file on a writable medium attached anywhere the kernel can find it, which can be a hard disk, USB-flash drive, or SD card. To store changes or updates permanently, this file can be used instead of the ramdisk (Knoppix

boot option `home=/path/to/knoppix.img`).

In the case that the *knoppix.img* file is present, but for some reason badly damaged, you will get an error message such as "cannot unite directories in `/UNIONFS`," but the system should still work – somewhat. You can rename or delete the defective image file.

Knoppix 4.x.x is quite old, which means that it will not recognize many SATA controllers. If no *knoppix.img* is present, this should not really cause an error message like the one above, unless you have physical read errors, which can well be caused by SATA if the DVD drive is connected to a SATA port and the corresponding driver puts the port in a wrong mode.

In this case, you cannot do much, other than wait for a new version with a newer kernel, try boot options for disabling DMA, or revert the port to compatible IDE mode in the BIOS settings of your computer.

Getting Connected



I need help because I'm a beginner with Linux. I have tried Knoppix 5.3.1, but I cannot get it to run on the Internet from one of the USB ports. (I have eight USB 2 ports.)

I don't have an ethernet port on my computer, which is a Dell Inspiron 530 with:

- router (with talk talk) SmartAX_MT882,
- 0.99 RAM,
- large hard drive.

Any help you can give me will be greatly appreciated.



The manufacturer does not seem to supply Linux drivers for USB mode, so my first choice would be to return the modem and get one that works.

My second choice would be to get a network adapter and connect with your router via LAN – it saves you a lot of headaches, and the router has a nice web-based configuration interface that helps you set up the Internet connection from Linux as well as Windows.

Send your Linux questions to
klaus@linux-magazine.com.