

Chapter 17

Hardware

This chapter introduces the user to various hardware devices of a system, and how devices may be added and configured.

Concepts Learned in this Chapter

- How to add and configure new hardware

Table of Contents

Hardware.....	1
17.1 Adding a Hard Drive	3
17.1.1 Physically Mounting Hard Drive.....	3
17.1.2 Partitioning Hard Drive.....	3
17.1.3 Formatting Partition.....	6
17.1.4 Assigning Directory to Partition	6
17.2 Configuring Raid	6
17.3 Configuring a Sound Card	6
17.4 Configuring a Video Card and Monitor	8
17.5 Configuring a Modem	8
17.6 Wireless Access	13
17.7 USB Devices	13
17.7.1 USB Memory Stick.....	13
17.7.2 External Hard Drive.....	13
17.7.3 Wireless NIC.....	13
17.7.4 Printer.....	13
17.7.5 PDA.....	13
17.8 1394 Firewire Interface.....	14
17.8.1 External Hard Drive.....	14
17.8.2 Camera Interface	14
17.9 Commands Used in this Chapter.....	14
17.10 Chapter Review Questions.....	14

17.1 Adding a Hard Drive

So you have an operational system – and you are running out of hard drive capacity – you need to add more. There are several issues that we need to address:

1. Physically adding an additional hard drive
2. Partitioning the hard drive
3. Formatting each partition
4. Assigning each partition to a directory location

17.1.1 Physically Mounting Hard Drive

It is assumed that the PC can support another hard drive. Recall that a PC can only support four IDE devices –

<u>Device Designator</u>	<u>System Designator</u>	
Primary	Master	hda
	Slave	hdb
Secondary	Master	hdc
	Slave	hdd

If you presently have one hard drive and a CDROM, then two devices have been assigned. During the POST at booting, you should be able to tell the assignment of each.

On the new hard drive, install the Berg plug in order to assign it as either a Master or Slave device. After you have the drive in place, insert the data cable into the connector – red strip next to the power cable. Then insert the power cable into its connector – red wire next to the data cable.

Now that the hard drive is installed, power up and check the BIOS – hit either the DEL or F1 key, depending upon the BIOS vendor. You will need to check that the system BIOS properly detects the new hard drive, that is, the drive is observed and its capacity is fully detected. Assuming all is OK, Save and Exit, and reboot. If not, go back and figure out what you did wrong (no answers provided here!).

17.1.2 Partitioning Hard Drive

OK, the drive is now installed and detected. Now we need to create partitions. This is accomplished using the **fdisk** utility. For this example, let's assume you have installed a new drive in the Secondary – Slave position. Start by issuing the command:

fdisk /dev/hdd

fdisk Application that partitions the drive
 /dev Device driver directory
 hdd Physical drive device driver (say that three times real fast ;-)

You now have the prompt:

Enter command (m for menu)>

Enter **m**

The most common options of interest that we need to use are:

- d Deletes partition – deletes what we created after we decided we decided we want to do it differently
- l List partition types, what we are particularly interested in will be:
 - 82 Linux Swap
 - 83 Linux native
 note that many other types of drive formats are also supported
- m Display this menu
- n Create new partition
- p List exiting partitions
- q Quit without saving
- t Change Partition type
- w Write Partition table to drive and exit

Now we need to create a new partition. Lets say we have an 80 Gbyte drive and will partition it with the following (your values will definitely be different):

- 1 20 G Linux
- 2 20 G Linux
- 3 15 G Linux
- 4 10 G Linux
- 5 10 G Linux
- 6 5 G Linux Swap

Recall that a hard drive can only support 4 physical partitions, one of which may be an extended partition that may contain additional logical partitions. The sequence of command for our example would be (® □ ENTER):

- n
- Partition Type: Primary / Extended p
- Partition Number: 1
- Start (default): (accept default)
- End (+XXM) +20000M
- n
- Partition Type: p
- Partition Number: 2
- Start (default): (accept default)
- End: +20000M
- n
- Partition Type: p
- Partition Number: 3

- Start (default): (accept default)
- End: +15000M
- n
- Partition Type: e
 - 1. ➤ Start (default): (accept default)
 - End (default): (accept default)
- p

<u>Device</u>	<u>Boot</u>	<u>Start</u>	<u>End</u>	<u>Blocks</u>	<u>ID</u>	<u>System</u>
/dev/hdd1		1	2550	20482812	82	Linux
/dev/hdd2		2551	5100	20482812	82	Linux
/dev/hdd3		5101	7013	15362109	82	Linux
/dev/hdd4		7314	10200	25000000	5	Extended

- n
- Start (default): ®
- End: +10000M
- n
- Start (default): ®
- End: +10000M
- n
- Start (default): ®
- End (default): ®
- p

<u>Device</u>	<u>Boot</u>	<u>Start</u>	<u>End</u>	<u>Blocks</u>	<u>ID</u>	<u>System</u>
/dev/hdd1		1	2550	20482812	82	Linux
/dev/hdd2		2551	5100	20482812	82	Linux
/dev/hdd3		5101	7013	15362109	82	Linux
/dev/hdd4		7314	10200	25000000	5	Extended
/dev/hdd5		7014	8288	10241406	82	Linux
/dev/hdd6		8289	9563	10241406	82	Linux
/dev/hdd7		9564	10200	5020703	82	Linux

By default, each new partition will be Linux (type 82), but this may be modified. We now need to modify the last partition to make it type Linux Swap. Issue the following:

- t
- Partition Number: 7
- Partition Type: 83

➤ p

<u>Device</u>	<u>Boot</u>	<u>Start</u>	<u>End</u>	<u>Blocks</u>	<u>ID</u>	<u>System</u>
/dev/hdd1	1	2550	20482812	82	Linux	
/dev/hdd2	2551	5100	20482812	82	Linux	
/dev/hdd3	5101	7013	15362109	82	Linux	
/dev/hdd4	7314	10200	25000000	5	Extended	
/dev/hdd5	7014	8288	10241406	82	Linux	
/dev/hdd6	8289	9563	10241406	82	Linux	
/dev/hdd7	9564	10200	5020703	83	Linux Swap	

If you try this with the MS fdisk application, you will find that it is considerably restricted compared to that available from Linux. Again, let's look at the partition types using the **t** option. Note that Linux can read not only Linux type partitions, but can also read and type to FAT and VFAT partitions, and read NTFS, OS-2 HTFS, and a multitude of other types.

17.1.3 Formatting Partition

Our hard drive partitions can now be referenced by the drive designator and partition number:

hdd1	hdd3	hdd5	hdd7
hdd2	hdd4	hdd6	

To format each partition, issue the following commands:

17.1.4 Assigning Directory to Partition

17.2 Configuring Raid

17.3 Configuring a Sound Card ¹

Linux supports a number of different sound cards – but not all. You can not buy a card and install drivers as you do in the Windows environment. Drivers must be installed per manufacture directions. Fortunately for the cards which are supported, the drivers are included in the installation process and may be

¹ Red Hat Linux Bible, by Christopher Negus; IDG Books; ISBN: 0-7645-4574-4

directly utilized. You can learn which cards are supported at www.4front.tech.com/ossfree/new_cards.html.

The following procedure will determine if the card can be installed and to configure it. This discussion starts with the assumption that the card is supported, but allows other cards to be configured if they are not on the acceptance list.

1. Before we start the configuration, we need to check what the IRQ and IO addresses are for the sound card. We are assuming here that the sound card is Plug-n-Play. Check the file **/proc/interrupts** and **/proc/ioports** for the settings of the sound card. Write them down for future reference.
2. Change to the **/usr/sbin** directory.
3. Issue the command:
sndconfig
4. There will be a request to perform a probe on your system for Plug-and-Play cards. Press **OK** to initiate the search. If the card is found, it will display the name and request if you want to configure it. As long as you are not using the “latest and greatest”, the card should be detected.
5. Before finalizing the configuration, a warning message is issued saying that the old **/etc/conf.modules** will be replaced (and backed up), press **OK**.
6. You are asked to play a sound file to test the configuration.
7. After you hear the sound, press **YES** and skip to step 13.
8. Since you did not hear the sound, it must be configured manually. Select **OK**.
9. A list of cards is displayed, select your card and click on **Enter**.
10. A list of IRQs, I/O Addresses, DMA and MPU settings are displayed. Refer to your card manual and select the proper settings. This requires you to know the settings of your card! After selecting the proper settings, click on **OK**.
11. You are warned that the **/etc/isapnp** file is being replace. Click on **OK**.
12. You are again asked to play a sound file for testing the card. Click on **OK**.
13. You are asked if you heard the sound. If you did not you need to select **NO** and repeat step 7.
14. It is assumed you have heard the sound file. You will test the card for MIDI. Click on **OK**.
15. If you heard the MIDI file, select **Yes**, otherwise select **NO**.

Linux supports a wide variety of sound file types, including:

8svx	gsm	txw	ub
aiff	hcom	vms	sb
au	mand	vox	uw
cdr	sf	wav	sw
cvs	smp	wve	ul
dat	Sunau	raw	

These file types may be converted between one another by using the **sox** utility.

Playing a Music CD

Of course, of prime interest in the CD is the ability to play a music CD. This is too simple. Two options are available.

On some installations, an icon may be visible which looks like a “music CD” under when your system is configured for the KDE desktop. Clicking on it will bring up the CD Player.

The second alternative is to access the music CD via the menu. ----

1. Click on the K at the bottom of the screen.
2. Select Multimedia
3. Select CD Player

In either alternative, from the menu, one should also open the Volume Control also under the K – Multimedia.

17.4 Configuring a Video Card and Monitor

17.5 Configuring a Modem ²

One of the most common communication interfaces for any computer system is the serial port, which in this example we will connect to a modem. Because most modems today are internal, some of this discussion may seem unnecessary; but if one needs to utilize the serial port for connecting to remote equipment, then we need to use the serial port. Use of the modem will allow one to establish a dialup connection to either another computer system or to the Internet.

There are two general types of modems available – internal and external. Although there are internal modems which are supported by Linux, it may be difficult to activate them due to possible special driver requirements. (In general you should generally need only the IRQ and the I/O settings.) Therefore for the general user it is the recommendation that an external modem be utilized. As an

² **Red Hat Linux Secrets** – 2nd ED; Naba Barkakati; IDG Books; ISBN: 0-7645-3175-1

additional benefit, an external modem typically has LED lamps as indicators of Send Data, Receive Data and Carrier Detect – which is an excellent status monitor.

As an aside, it is recommended that a computer system that utilizes an external modem be equipped with two serial ports – one for the modem and a second for AC Power monitoring if a UPS system is employed.

It is assumed that the user has sufficient background to properly connect an external modem to the computer. For this discussion, we will assume that COM1 at IRQ 4, I/O address 0x3f8 (hex 3f8) is utilized. A different COM port will require the user to confirm the IRQ – I/O assignments.

There are two different port name assignments within the Unix (Linux) when we wish to communicate with the COM1 port – **ttyS0** and **cua0**. **ttyS0** is designated as the incoming COM Port – that is data flows into the computer. **cua0** is designated as the outgoing COM Port, data flow out of the computer. Latter configurations of Linux commonly provide bi-directional status to **ttyS0**.

Outgoing Call Setup

There are a number of standard **AT** commands that were originally established by Hayes modems. The modem that you utilize may support an enhanced set, in which case you must refer to the modem documentation. Some of the more common commands include:

Configuration:

ATE	Echo
ATL	Speaker volume
ATM	Speaker control
ATQ	Quiet mode
ATV	Verbose mode
ATX	Result codes
AT&V	Stored Profile

Action:

ATDP	Dial – Rotary / Pulse
ATDT	Dial – Touch Tone
ATDL	Redial last number
ATH	Hook control
ATA	Answer incoming call
ATO	Online status
ATZ	Software reset
AT&F	Factory default settings
ATSn	A set of specific configuration registers

In order for any user besides the ROOT administrator to be able to dial out through the com port, the ttyS0 access must be opened. The default does not provide this, so issue the command from the **/dev** directory:

```
ls -l ttyS*
chmod o+rw /dev/ttyS0
ls -l ttyS*
```

Note the difference between the two ls displays.

Linux includes a program that provides basic terminal functionality, called **minicom**. This emulates a VT102 “dumb terminal”. To activate the terminal program, issue the command either at the shell (CLI) prompt or in a xterm window:

minicom

For those familiar with the DOS Telex or Procomm communication programs, minicom is very similar.

The first time this is done, an error message may be observed stating that there is no global configuration file. As the ROOT administrator, issue the command:

minicom -s

You will have a screen something like the following:

--> minicom screen

The most important settings for the serial setting is data rate, bits/character, parity, and stop bits.

Data Rate	In order to communicate with a remote device, the data rate must be the same. Otherwise data errors will occur.
Bits/Character	We transmit using either 7 or 8 bits per character. If using 7 bits, then parity is typically employed and only the alphanumeric character set is transmitted, or 128 characters. If 8 bits are employed are employed, then additional characters may be transmitted, allowing a total of 256 characters.
Parity	Parity provides a means for error detection. Each character has an additional bit added to it, allowing for either even or odd parity, meaning that if the original data has an even number of one's and parity is set to odd, then the parity bit is set to a 1, providing a total number of odd ones.
Stop-Bit	The Stop-Bit provides a minimum separation between characters. Options include None, 1 Stop Bit, 1 ½ Stop Bits, and 2 Stop Bits. Typical setting is for 1 Stop Bit.

This allows the setup of **minicom** to communicate with the desired COM Port and to the modem. These must be properly set in order for us to establish communications with the modem – which is necessary before we can communicate to an external system. Some of these settings may require

reference to the modem manuals. Generally one can accept the default settings, but the one setting most likely requiring a change will be the Initialization String.

In addition, the following must be performed:

1. Edit the **/etc/minicom.users** file. Add the line to the end of the file with the single word **ALL** in order to enable all users to access the **minicom** configuration file (this may already exist).
2. Create a link between the **/dev/modem** and the serial port device (where it is connected). Issue the following command:
ln -s /dev/ttyS0 /dev/modem

We are now ready to test communications between the computer and the modem. After starting **minicom**, issue the command:

AT

and you should obtain the response of:

OK

You are now in communication with the modem.

To dial a remote system, such as another system at 800-555-1212, issue the command:

ATDT 18005551212

Incoming Call Setup

In order to support an incoming modem call to our system, we need to set up several configurations within Linux. This provides a login prompt.

We need to first edit (create) the file **/etc/rc.d/rc.serial**. If using a 56 Kbps modem, include the line:

/bin/setserial /dev/ttyS0 spd_vhi

Assuming that the modem supports RTS (Request to Send) and CTS (Clear to send) hardware flow control handshaking, include the command:

/bin/stty crtscts < /dev/ttyS0

Next the **uugetty configuration** file must be updated. This will supply the **login** prompt.

Edit the **/etc/gettydefs** file to something like the following:

```
#38400 fixed – brand modem entry
F38400# B38400 CS8 CRTSCTS # B38400 SANE -ISTRIP HUPCL #@S
Login: #F38400
```

Where:

F38400	is a label
CRTSCTS	enables hardware flow control

#@ codes

@B	display current data rate
@D	display current date

@L	display serial port designator (ttyS0)
@S	display system name
@T	display current time
@U	display the number of currently logged in users
@V	display the uugetty version number

Edit the **/etc/default/uugetty.ttyS0** file to include the following (comments not shown):

```
INITLINE = cua0
ALTLOCK = cua0
TIMEOUT = 60
INIT = "" \d+++ \dAT\r OK\r\n ATH0\r OK\r\n
AT\sM0\sE1\sQ0\sV1\sX4\sS0\s&C1\s&S0\r OK\r\n
WAITFOR = RING
CONNECT = "" ATA\r CONNECT\s\A
DELAY = 1
```

Most of these settings should be default. This allows **uugetty** to initialize the modem, wait for a call, and then issue the login prompt.

The INIT line initializes the modem by:

1. Issue AT command
2. Look for OK reply
3. Issue On Hook (ATH0)
4. Look for OK reply
5. Turn off the speaker (ATM0)
6. Turn on Echo (ATE1)
7. Set Quiet Mode (ATQ0)
8. Set Verbose Mode (ATV1)
9. Provide detailed results (ATX4)
10. Auto Answer Off (ATS0=0)
11. Control Carrier Detect (AT&C0)
12. Turn on DSR (Data Set Ready) (AT&S0)
13. Look for OK reply

When an incoming call is initiated, the RI (Ring Indicator) line goes high and a CONNECT is initiated. An ANSWER command (ATA) is then issued after the specified number of rings.

After the call is answered, the contents of the **/etc/issue** file are transmitted, followed by the **login** prompt. Then **uugetty** runs the **/bin/login** program to process the user login response for username and password.

Finally, we need to set up the serial port so that it is initiated during the boot process. We need to edit the **/etc/inittab** file to start a copy of **uugetty** on the **/dev/ttyS0** line. Within the **inittab** file, find the lines:

```
#Serial line with dial-in modem
s0:235:respawn:/sbin/uugetty ttyS0 F38400 vt100
```

The arguments of this command line are:

235	initiates uugetty at run levels 2,3,5
ttyS0	specifies the serial port 0 (COM 1) for which uugetty must monitor
F38400	specifies the line speed, this must also appear in the /etc/gettydefs file
vt100	specifies the monitor type, this must be the same as appears in the /etc/termcap file

Testing the Dial-In Setup

After the various files have been updated, we need to restart the **/etc/inittab** file. As ROOT, issue the command:

```
init q
```

If you have the ability to dial into your system, or have a second terminal which can be connected to the serial port, you should have something like the following:

```
Red Hat Linux release 5.0 (MyPC)
Kernal 2.0.31 on an i586
```

```
sysname login:
```

Your system should now be operational.

17.6 Wireless Access

17.7 USB Devices

17.7.1 USB Memory Stick

17.7.2 External Hard Drive

17.7.3 Wireless NIC

17.7.4 Printer

17.7.5 PDA

17.8 1394 Firewire Interface

17.8.1 External Hard Drive

17.8.2 Camera Interface

17.9 Commands Used in this Chapter

17.10 Chapter Review Questions

Chapter Index

	A			
Adding a Hard Drive		3	Partitioning	3
	C		Physically Mounting	3
Configuration			Primary	3
Modem		8	Master	3
USB		13	Primary Slave	3
Wi-Fi		13	Secondary	3
Configuring			Master	3
Raid		6	Slave	3
Sound Card		6		
Video Card and Monitor		8	U	
	H		URL	
Hard Drive			www.4front.tech.com/ossfree/new_c	
Assigning Directory		6	ards.html	7
Formating		6	Utility	
			fdisk	3