Linux

Timothy Daly

February 24, 2004

Abstract

This class covers Linux in three parts. First we detail the operating system called Linux and outline what it does. Next we detail the Linux distributions and show how to install and configure them. Finally we detail the Linux social and business environment and discuss some of the current community issues.

Contents

1 Weekly News

Pamela Jones (Groklaw) [1] got a job by giving away her expertise.

Pamela Jones has been running the Groklaw website for about a year coordinating and collecting information on the SCO–IBM lawsuit. This has been a purely volunteer effort on her part and has been very useful in focusing the open source community in fighting this challenge. As a result of her collaboration with Daniel Egger at Open Source Risk Management (OSRM) she made an industry contact. Last week (Feb 3, 2004) she announced:

> OSRM has simultaneously retained me, part-time, to work on their indemnification project as their Director of Litigation Risk Research.

Axiom and TechExplorer: How Money Slants the Thinking: When Axiom was a commercial product it had a front end called TechExplorer. The front end was not released as part of the open sourcing of Axiom. Instead it was sold to a company. One of the original authors left IBM and followed the product. Given that he was a co-author of Axiom I suggested that we do joint work to ensure that Axiom and TechExplorer worked together. He thought this was a great idea.

There was one "sticking point", however. Since TechExplorer is a commercial product I would be working on the interface for free. That means that I'd be giving away work to a company that would ultimately sell the work for profit. Yet I would not be paid.

However, you have to remember that open source software development is a gift economy. You have to be prepared to give away your time and talents to all. The ultimate benefit is that all Axiom users and all TechExplorer users have more useful tools.

Remember: It's what you do for free that matters most.

James Blackwell and Gnu-Arch: On Sunday night of this week (Feb 8, 2004) I ran into a problem with the Gnu-Arch [3] software version control system. I went to the website and found that there was an IRC (Internet Relay Chat) channel for Gnu-Arch [4]. I signed onto the IRC server and joined the channel. I posted my problem.

James Blackwell [5] was online in the chat room at the time. He diagnosed the problem and told me what I did wrong. He went even further and gave me a short tutorial which included the proper sequence of commands that I needed to solve my problem. **This** is the real power of open source programming. There is no commercial product I'm aware of that will provide free debugging service at 1 A.M.

County Energy bets business on open source [**9**] has switched from using Irix and Tru64 to Linux. They use a combination of open source

such as Apache and proprietary software such as WebLogic and Lotus Notes. The article quotes him as saying:

> "Talk to your peers about open source as there is not really a downside," he said. "You can use it without risk and it won't cost you anyhing other than a bit of time. You'd be mad if you didn't try it."

Eben Moglen and Open Source Development Labs have issued a position paper on the SCO-IBM lawsuit and its impact on Linux [10]. In their press release they say:

> "We see Linux deployments continuing around the world and many prudent customers are choosing to ignore SCO's legal threats until the courts rule, particularly given SCO's admitted uncertainty about its own rights" [11]

The Open Source Development Labs is a joint venture, non-profit organization that provides Linux expertise and computing and test facilities. OSDL members include: Alcatel, Cicco, Co-Create, Computer Associates, Dell, Ericsson, Force Computers, Fujitsu, HP, Hitachi, IBM, Intel, Linuxcare, Miracle Linux Corporation, Mitsubishi Electric, MontaVista Software, NEC Corporation, NEC Soft, Network Appliance, Nokia, NOvell, NTT COMWARE, NTT DATA INTELLINK, Red Hat, Sun Microsystems, SUSE LINUX, TimeSys, Toshiba Solutions, Transmeta Corporation, Trolltech, Turbolinux, Ulticom, Unilever, VA Software, and Wind RIver Systems.

Hopefully that will convince you that there are some job opportunities with companies related to open source programming.

2 Linux in the Small

Linux in the small refers to the Linux Kernel. That is, the operating system that sits below the user space programs and controls the machines. The Linux Kernel is a very active group of developers. You can see a summary of the weekly activity on the Linux Kernel Mailing List [6].

The mailing list is where the commmunity of kernel developers discuss and argue about various development points. Linux is the largest known example of an open source project having several hundred developers.

2.1 Developers

Individual developers work on a particular piece of code. For example, you could be an expert in how your digital camera works. So you might work on the USB device driver that handles your camera.

Maintainers work on subsystems. There is a list of kernel maintainers [8]. They handle such things as the quota subsystem, the block device drivers, the PPC64 architecture, etc. The whole kernel is broken up into areas of responsibility. Maintainers accept changes (patches) from individual developers, apply them, test them, and integrate them into larger changesets to be forwarded upward.

Linus Torvolds is the top of the development pile. He only accepts changes that have come thru the appropriate maintainers and then only if he agrees with them. Thus he controls what goes into the official kernel. There are people, like Andrew Morten, that maintain their own "branch" of the kernel. The Morten kernel can be built using the "-mm" switch which includes and/or excludes certain things that Linus has not accepted.

2.2 Operating System

Linux is an operating system. It has the goal of managing the resources of your computer. In particular, if you look at the source code [7] you'll find that it breaks down into several parts, thus:

initialization is the process that happens when you power-on your computer. The kernel initialization code parses the command line options, calibrates the speed of the machine, initializes various tables, and starts the first "program" running (called init). The init program then reads configuration files and starts other programs. (see [7] pp443–450).

system calls are a set of services that the operating system provides to user programs. System calls allow you to do things like read the time from the hardware clock. Since the hardware clock is shared by all running programs access to the clock has to be managed. (see [7] pp451–458)

signals, interrupts, and time are the primary tools to allow the operating system to make it appear that there are multiple programs running in parallel. The resources of the computer are time-sliced and scheduled in very small increments to give the impression of many things happening at once. (see [7] pp459–478)

processor or CPU (central processing unit) is managed by the operating system and is allocated among the many processes and threads. Linux maintains priorities and schedules use of the processor. (see [7] pp479–498)

memory in a computer forms a hierarchy. There are registers within the CPU, two or three levels of cache where level 1 cache is faster than level 2 cache. Level 2 cache is faster than level 3 cache. Beyond the cache is slower main memory. And beyond main memory is the hard drive which is the slowest of all the devices (see [7] pp499–520) (**refrigerator story**).

devices that are attached to the computer are next. Examples of devices are your keyboard, screen, speakers, PCMCIA cards, etc. Since access to these devices need to be shared the operating system manages them. Linux has a fairly uniform way of handling most device drivers that are dynamically loaded as needed. (**tables story**)

3 Linux in the Medium

Linux also refers to more than just the operating system. Many companies put together Linux "distributions". There are many thousands of open source projects. Each of these projects is packaged up into installable file formats such as .tgz (tar gzipped), RPM (Redhat Package Manager), or apt (debian apt-get) format. Since there are literally hundreds of editor and dozens of screen managers it is very difficult and time consuming to find, download, install, and configure compatible sets of tools. The distribution companies have taken on the task of packaging a set of tools that work together into simple installable CD sets.

3.1 Distribution Philosophy

Linux distributions differ in the philosophy they use to guide their respective choices. Some examples are:

RedHat strives for "best of breed" packaging. They try to find the best available packages in various categories (e.g. editors). Then they choose one or a few. Finally the build a set of CDs which contains both the binary executables and the source code. Redhat has recently announced that they will no longer be shipping general purpose CD sets and have given that project over to the open source community under the name Fedora. Redhat has decided to specialize in "enterprise-scale" linux systems.

Debian attempts to package the world for the world. There are several thousand packages in the debian distribution. The debian maintainers also have facilities to compile each package for several different machines including 64bit hardware and ARM (Palm Pilot) processors. So you not only get a program that runs on a PC but one you can run on your Zaurus handheld. Debian is a "purist" kind of distribution in philosophy. They will not accept non-free software. They do distribute binaries. They do not package a full distribution but expect you to download it.

Gentoo works directly from sources. You compile things from scratch. In general, this distribution is also "purist" and will not accept non-free software.

SuSE is semi-commercial. They charge for their distribution and do carry non-free software.

Knoppix is designed to run on any hardware without installing. It is a great tool for (a) testing if Linux will run on your hardware, (b) recovering a broken linux installation, and (c) installing a large, easy to use, fully configured system.

3.2 Installing Mandrake

The Mandrake operating system will be installed during class. We outline the steps involved here.

3.2.1 BIOS setup

The BIOS (Basic Input/Output Subsystem) is a piece of software residing on a Flash ROM (software-changeable Read-Only Memory). The BIOS settings can usually by reached by hitting F2. The main change that needs to be made is the hardware boot sequence.

Normally the hardware tries to find the operating system on the hard drive. If that fails it tries the CDROM and then the floppy drive. In order to install a new system you need to change the order so that the system tries to read from the CDROM first.

3.2.2 Setup vs Upgrade

With the same set of CDROMs you can either set up a new system or upgrade a previously setup system. We're only going to walk thru the new system setup here. The goal is to get a working copy of the Linux operating system and other tools installed on the hard drive so that the computer will start Linux next time it is powered on.

3.2.3 Language

As the open source effort is worldwide you will be asked for a language choice. Most systems default to American English but not all. Red Flag Linux, from China, defaults to Mandarin.

One place that you can have an immediate impact on the world is to use your language skills to change the messages used by a piece of software to other languages.

3.2.4 License

You will be asked to agree to a License. The main points of most open source license are to limit the liability of the software distributors, to inform you of the type of license (usually GPL but not always), to make sure that you understand that the copyrights reside with the authors, to describe what laws of what jurisdictions might apply, and to provide full employment for lawyers.

3.2.5 Hardware Detection

Most distributions can detect features of most hardware configurations. Some items may not be detected, such as mice with wheels, and need to be explicitly named. As Linux has the task of managing this hardware it is important that your choices are compatible with the real hardware.

3.2.6 Security

Linux has several security options including built-in firewall protection. There are pre-configured firewall rules that will disallow various kinds of network connections. There is a tradeoff of security versus functionality. For instance, if you set up a really secure firewall it is unlikely that you will be able to ftp files from one system to another as ftp is known to be insecure and the firewall will block it.

3.2.7 Disk Partitioning

Traditionally Linux (and Unix) hard drives are managed as though the were made up of smaller hard drives, called partitions. Partitioning allows you to decide where you want to install items on the disk. We choose custom disk partitioning so we can specify the size and location of these drives.

There are a couple important choices that need to be made. You need to specify the location, size, mount point, and type of file system.

First we need a /boot partition. This is where Linux images (copies of the linux system that get started on power-on) are kept. This is generally small and on our test system we choose to allocate this first with a size of 98MB. We put this first on the full hard drive and it gets the device name (/dev/hda1). We choose the label (/boot) and the ext3 filesystem.

Next we need a / partition. This is where Linux keeps everything including all of the files you normally see. This is generally large (usually a max of about 5GB but we only have a 2GB drive) and we make this second with a size of 1.7GB. We put this second on the full hard drive and it gets the device name (/dev/hda2). We choose the label (/) and the ext3 filesystem.

Normally the next partition would be /home. User directories are traditionally kept under /home and installations of Linux will not normally change or format this area. If you use /home it is possible to upgrade or even change Linux distributions and have user data survive the complete change. Due to a lack of hard drive space we skip this for demonstration purposes.

Next we need to allocate swap space. This is where the operating system will find space to "page" running programs. It is temporary space managed by the operating system so it can load and run more programs than will fit in main memory. We generally allocate swap space to be twice the size of main memory. In this case we choose a size of 169MB. We put this third on the full hard drive and it gets the device name (/dev/hda3). We don't need to label this space and it has a "swap" file format.

3.2.8 Drive Format

Hard drives are basically just a set of blocks of bits. There are two kinds of "format" programs that manipulate these blocks. One, usually done at the factory, is a "low-level" format which writes lead-in, data, and lead-out information as well as marks bad blocks so they are not used.

The second kind of "format" program is the one used during install. It sets up the file system format (allocates inodes) and file system size/type information. Both types of format erase what was previous on the drive (well, not really but you normally don't care).

3.2.9 Package Group Selection

Computers have uses that fall into different categories and the software they have installed also falls into different categories. Package Group Selection tries to classify the computer use (Workstation, Server) and the groups of tools each user might want. There are thousands of individual packages and some packages require other packages. The Mandrake distribution has figured out these groupings and dependent relationships and made installation much easier.

Once we make choices the program tries to decide if all of the dependent software is included and, if it is, proceeds to expand and copy our selections to the hard drive root partition.

3.2.10 Setting the root password

Next we set the root password, hopefully to something secure. No, your dog's name is not a secure password.

3.2.11 Setting up user accounts

Set up a user account, at least for yourself. You should not work as root except to install new software. Did I mention that your dog's name is not a secure password?

3.2.12 Installing a boot loader

We now save out a runnable linux image to the /boot partition. When we power-on the system the BIOS will look in the MBR (master boot record) to find the disk address of the boot loader. The boot loader will find the disk address of the linux kernel image. It will copy this image into memory and transfer control to that image. The linux kernel will start the initialization process mentioned elsewhere in this document.

The system will ask you where you want the new boot loader installed. You have the choice of the MBR (master boot record) or the first sector of the root partition. If you are only running one system on a computer choose the MBR. If you are dual-booting the system choose the first sector of the root partition.

3.2.13 Configuring the install

Most of the installation has already been configured based on choices you've made. In Mandrake the Graphical Interface is not configured by default and you should configure it. We won't do that here as this is just an example.

3.2.14 Updating packages

Since we just installed everything we don't need to update anything. If we restart the computer with the boot CDROM in the drive we can upgrade the system at any time.

3.2.15 Exit install

Remove the CDROM and reboot the system. You should now have a running Linux.

3.3 Installing Knoppix

KNOPPIX is a system that will boot directly from CD and not touch the hard drive. Thus it can be booted on Windows machines without damaging Windows.

KNOPPIX (as of version 3.3) will also allow you to install a disk image. To do this you need to open a command shell and type: kxn-install. The KNOPPIX CD contains 2.2GB of compressed data and these will be copied to the hard drive in a bootable format.

KNOPPIX is based on Debian so once you have it installed you can use apt-get to download and install new packages.

3.4 Configuring Linux

3.4.1 Network configuration

Once you have Linux running on a machine you need to get it connected to the network. This is a fairly complicated topic. You can enable DHCP which will automatically choose an IP address for your machine if the network you plug into supports DHCP (most do). If not you have to obtain an IP address from somewhere. Assume you have the address 10.0.0.5 and that the network gateway is 10.0.0.1; the network hardware port is at eth0. You can start up your connection by typing (as root):

```
ifconfig eth0 10.0.0.5 netmask 255.255.255.0 up
route add default gw 10.0.0.1
```

at which point you can probably talk to other machines over the net. You may have to configure a nameserver by editing (as root) the file (/etc/resolv.conf). It will look like:

```
nameserver 198.32.42.38
nameserver 198.32.42.39
```

3.4.2 Process enabling

There are a large number of processes that can be enabled or disabled at various "run levels". Run levels are stages of system booting that start various processes. The levels exist because certain processes require other to exist before they start. For instance, you can't start networking programs until the network is started. But you can't start the network until the network driver is loaded. But you can't load the network driver until you detect the hardware.

Linux systems generally run in either level 3 or level 5. Level 3 is a "console mode". That is, it presents a single console to the user. Level 5 is "X mode". That is, the user interface is generally X or some other GUI handling program.

There is a command "chkconfig –list" which will show you all of the processes and the level at which they are on or off.

```
bash-2.05b# chkconfig --list
kudzu           0:off 1:off 2:off 3:on 4:on 5:on 6:off
syslog          0:off 1:off 2:on 3:on 4:on 5:on 6:off
netfs           0:off 1:off 2:off 3:on 4:on 5:on 6:off
network         0:off 1:off 2:on 3:on 4:on 5:on 6:off
random          0:off 1:off 2:on 3:on 4:on 5:on 6:off
rawdevices      0:off 1:off 2:off 3:on 4:on 5:on 6:off
pcmcia          0:off 1:off 2:on 3:on 4:on 5:on 6:off
saslauthd       0:off 1:off 2:off 3:off 4:off 5:off 6:off
keytable        0:off 1:on 2:on 3:on 4:on 5:on 6:off
apmd            0:off 1:off 2:on 3:on 4:on 5:on 6:off
atd             0:off 1:off 2:off 3:on 4:on 5:on 6:off
gpm             0:off 1:off 2:on 3:on 4:on 5:on 6:off
autofs          0:off 1:off 2:off 3:on 4:on 5:on 6:off
iptables        0:off 1:off 2:off 3:off 4:off 5:off 6:off
irda            0:off 1:off 2:off 3:off 4:off 5:off 6:off
isdn            0:off 1:off 2:off 3:off 4:off 5:off 6:off
sshd            0:off 1:off 2:on 3:on 4:on 5:on 6:off
portmap         0:off 1:off 2:off 3:on 4:on 5:on 6:off
nfs             0:off 1:off 2:off 3:off 4:off 5:off 6:off
nfslock         0:off 1:off 2:off 3:on 4:on 5:on 6:off
sendmail        0:off 1:off 2:off 3:off 4:off 5:off 6:off
rhnsd           0:off 1:off 2:off 3:on 4:on 5:on 6:off
crond           0:off 1:off 2:on 3:on 4:on 5:on 6:off
anacron         0:off 1:off 2:on 3:on 4:on 5:on 6:off
httpd           0:off 1:off 2:off 3:off 4:off 5:off 6:off
aep1000         0:off 1:off 2:off 3:off 4:off 5:off 6:off
bcm5820         0:off 1:off 2:off 3:off 4:off 5:off 6:off
squid           0:off 1:off 2:off 3:off 4:off 5:off 6:off
tux             0:off 1:off 2:off 3:off 4:off 5:off 6:off
winbind         0:off 1:off 2:off 3:off 4:off 5:off 6:off
smb             0:off 1:off 2:off 3:off 4:off 5:off 6:off
xfs             0:off 1:off 2:on 3:on 4:on 5:on 6:off
```

```
xinetd          0:off 1:off 2:off 3:on 4:on 5:on 6:off
cups            0:off 1:off 2:on 3:on 4:on 5:on 6:off
named           0:off 1:off 2:off 3:off 4:off 5:off 6:off
ntpd            0:off 1:off 2:off 3:off 4:off 5:off 6:off
firstboot       0:off 1:off 2:off 3:off 4:off 5:off 6:off
snmpd           0:off 1:off 2:off 3:off 4:off 5:off 6:off
snmptrapd       0:off 1:off 2:off 3:off 4:off 5:off 6:off
pxe             0:off 1:off 2:off 3:off 4:off 5:off 6:off
vsftpd          0:off 1:off 2:off 3:off 4:off 5:off 6:off
yppasswdd       0:off 1:off 2:off 3:off 4:off 5:off 6:off
ypserv          0:off 1:off 2:off 3:off 4:off 5:off 6:off
ypxfrd          0:off 1:off 2:off 3:off 4:off 5:off 6:off
dhcpd           0:off 1:off 2:off 3:off 4:off 5:off 6:off
dhcrelay        0:off 1:off 2:off 3:off 4:off 5:off 6:off
xinetd based services:
chargen-udp: off
rsync: off
chargen: off
daytime-udp: off
daytime: off
echo-udp: off
echo: off
services: off
servers: off
time-udp: off
time: off
cups-lpd: off
sgi_fam: on
finger: off
rexec: off
rlogin: off
rsh: off
ntalk: off
talk: off
telnet: off
```

3.4.3 Tuning

You can change device parameters on your system to improve performance. For instance, there are tools such as hdparm which will allow you to measure the performance of your hard drive along with options to allow you to change performance parameters.

```
bash-2.05b# hdparm -tT /dev/hda

/dev/hda:
```

```
Timing buffer-cache reads:   128 MB in  0.36 seconds =355.56 MB/sec
Timing buffered disk reads:  64 MB in  2.25 seconds = 28.44 MB/sec
```

3.5 UML

Besides running Linux on the native hardware it is possible to build a program that acts just like the hardware but is a simple user program. If you had such a "virtual hardware program" you could run Linux inside that program. This ability already exists and is called UML (User Mode Linux) [13].

User mode linux will allow you to build new systems and experiment with changing them or allow you to debug them without having extra hardware. It will also allow you to create a whole network of systems within one computer so you can experiment with things like beowulf clusters without having the hardware available. As processors become more powerful, more memory becomes available, and hard drive space improves it will be possible to have multiple virtual computers running on one real computer. This is not a new idea and was used on the IBM Mainframe computers in the 1960s (the IBM VM operating system).

4 Linux in the Large

Linux is more than an operating system or a distribution. It is an idea. The idea is that you should be able to read, write, and run your own software on your own machines for your own purposes. You are not restricted to buy anything from anyone. This has the potential to make everyone able to help everyone else. In a gift economy, when you help the community, "like a rising tide, all boats rise".

4.1 Countries

Whole countries are experimenting with Linux and several have started efforts that will result in their own distributions. China, South Korea, and Japan [14] have agreed to build a Linux distribution that will be used by the people of that region. This amounts to about one-half the population of the world. It appears that the Red Flag Linux distribution will be the basis for this effort.

This kinds of support makes it possible to standardize across large regions of the world without paying a "Microsoft tax"; that is, without forcing people to buy Microsoft in order to talk to the world.

4.2 Governments

Peru is considering mandating the use of Open Source. There is a famous letter [15] from Dr. Nuñez, Congressman of the Republica of Peru, that lays out the reasons why government sites have an obligation to use free rather than proprietary software.

Again, this is a case where people should not be forced to buy Microsoft in order to visit their government websites. In England, at the moment, you must use Microsoft software to partake in certain government services.

4.3 Businesses

Refer to the News section above to see an example of a business that considers Linux as a useful tool.

Businesses are being forced to upgrade their software. Microsoft has required all new software purchases to come with a license that requires periodic upgrades (on Microsoft's schedule). Further, Microsoft has dropped support for older versions of Windows (3.1, 95, 98, NT, 2000). This leaves businesses trapped into either using unmaintainable software or upgrading even though it will severely impact the business. Linux gives you the freedom from this.

4.4 Schools

Linux makes a perfect operating system for schools. It allows students to study the actual code used. Students can change the code to test out new ideas without starting from scratch. Students can freely copy the operating system to any and all machines they own without fear of violating license agreements.

5 Linux Community Issues

Linux, viewed as a community, has a set of issues which are currently part of the zeitgeist ("in the air"). You should get familiar with these discussions as they motivate a lot of the discussions.

5.1 Patents

Patents are a serious threat to open source software. If I, as the copyright holder of a newly written piece of software, license the software under the GPL then I give everyone the permission to use it. No company can lock up the software as proprietary and no company can prevent you from using it. Patents have that power.

An example of where patents can cause a great deal of harm is the .gif format files. Gif files were widely used for years. Unisys discovered that it had a patent that covers the .gif format [16]. In particular, Unisys writes:

> Unisys has frequently been asked whether a Unisys license is required in order to use LZW software obtained by downloading from the Internet or from other sources. The answer is simple. In all cases, a written license agreement or statement signed by an authorized Unisys representative is required from Unisys for all use, sale or distribution of any software (including so-called "freeware")

and/or hardware providing LZW conversion capability (for example, downloaded software).

The .gif format uses LZW compression. You can't use this technique without a written license from Unisys. Even if you discovered the same technique on your own you could still be sued and your program would have to be withdrawn from sale.

5.2 Desktop

Linux has been widely accepted as excellent software for servers. There are many companies that sell Linux for this purpose. In fact Red Hat, the leading distributor of Linux, has turned its whole business to focus on selling "Enterprise" class systems for servers.

Yet the place most people touch computers is on the desktop. Everyone expects that they will have the tools available that currently are shipped with Windows. There are two basic issues.

The first issue is making tools which are compatible with the Windows programs. That is, tools which will read and write Microsoft file formats. XML is an attempt to standardize the file formats so information is not locked up in things like .doc files. Microsoft, in order to fight this threat, is patenting its Windows file formats.

The second issue is making Linux accessible to the average user. It is claimed that Linux is hard to install. Windows installations are just as hard but they are done by the manufacturer. Dell and others have started campaigns to sell pre-installed Linux systems but Microsoft has threatened their business (by refusing to license Windows) and the manufacturers have all withdrawn from their attempts. As you can see from this class the KNOPPIX distribution makes installations nearly painless.

5.3 SMB

Another issue is the SMB (Small and Medium Business) area. The Fortune 1000 companies are pretty well saturated with software and have large IT departments to handle the software. But small (¡50) and medium (¡500) businesses usually cannot support a staff to handle software. Thus it is important to have software with a low-overhead in terms of staff, cost, and time.

There is a battle for mindshare about the subject of TCO ("Total Cost of Ownership") which includes everything about computers (staff, space, hardware, software, licensing, training, wiring, upgrades, etc). Microsoft, quite correctly, claims that the cost of the software is a very small portion of the total cost of TCO. Thus it should not matter if the software if free. What they fail to mention is that Microsoft software costs more in terms of staff training, desktop and website downtime, licensing, etc. Much ink is being wasted to convince SMBs that Microsoft is cheaper and more reliable than Linux. It's not true but that has never stopped the marketing machine.

5.4 Virus, Worms, Trojans, and Spam

The four horsemen of the internet continue to spread doom and gloom. Governments are getting involved. There is a famous paper by Geer, et. al. [17] which warns of the "security dangers posed by software monopolies" and the effects of a software "monoculture".

Stopping the spread of these problems can be done in one of two ways. We can conduct a massive campaign to educate people how to handle these threats, how to keep their systems secure, and how to behave in order to minimize the damage. This path is prone to failure.

The second path is to make the software more secure so that these threats do not have an environment where they can flourish. Windows is not that environment for many reasons.

6 Appendix A: KNOPPIX cheat codes [2]

```
                    CHEATCODES AND HINTS FOR KNOPPIX V3.3
================================================================================
                          (last update: 22.09.2003)

These options (can be combined) work from the SYSLINUX bootprompt:

knoppix lang=cn|de|da|es|fr|it|nl   specify language/keyboard
knoppix lang=pl|ru|sk|tr|tw|us      specify language/keyboard
knoppix alsa (or alsa=es1938)       Use ALSA sound driver (at your own risk)
knoppix desktop=fluxbox|icewm       Use specified WM instead of KDE (1)
knoppix desktop=kde|larswm|twm      Use specified WM instead of KDE (2)
knoppix desktop=wmaker|xfce         Use specified WM instead of KDE (3)
knoppix screen=1280x1024            Use specified Screen resolution for X
knoppix xvrefresh=60 (or vsync=60)  Use 60 Hz vertical refresh rate for X
knoppix xhrefresh=80 (or hsync=80)  Use 80 kHz horizontal refresh rate for X
knoppix xserver=XFree86|XF86_SVGA   Use specified X-Server
knoppix xmodule=ati|fbdev|i810|mga  Use specified XFree4-Module (1)
knoppix xmodule=nv|radeon|savage|s3 Use specified XFree4-Module (2)
knoppix xmodule=radeon|svga|i810    Use specified XFree4-Module (3)
knoppix 2                           Runlevel 2, Textmode only
knoppix floppyconfig                Run "knoppix.sh" from a floppy
knoppix myconf=/dev/sda1            Run "knoppix.sh" from a partition
knoppix myconf=scan (or config=scan) Try to find "knoppix.sh" automatically
knoppix home=/dev/sda1/knoppix.img  Mount loopback file as /home/knoppix
knoppix home=scan                   Automatic search for knoppix homedir
knoppix no{apic,agp,apm,audio,ddc}  Skip parts of HW-detection (1)
knoppix no{firewire,pcmcia,scsi}    Skip parts of HW-detection (2)
knoppix no{swap,usb}                Skip parts of HW-detection (3)
failsafe                            Boot with (almost) no HW-detection
knoppix pci=irqmask=0x0e98          Try this, if PS/2 mouse doesn't work *)
knoppix pci=bios                    Workaround for bad PCI controllers
knoppix ide2=0x180 nopcmcia         Boot from PCMCIA-CD-Rom (some notebooks)
knoppix mem=128M                    Specify Memory size in MByte
knoppix dma                         Enable DMA for ALL IDE-Drives
knoppix noeject                     Do NOT eject CD after halt
knoppix noprompt                    Do NOT prompt to remove the CD
knoppix vga=normal                  No-framebuffer mode, but X
knoppix blind                       Start Braille-Terminal (no X)
knoppix brltty=type,port,table      Parameters for Braille device
knoppix wheelmouse                  Enable IMPS/2 protocol for wheelmice
knoppix nowheelmouse                Force plain PS/2 protocol for PS/2-mouse
fb1280x1024                         Use fixed framebuffer graphics (1)
fb1024x768                          Use fixed framebuffer graphics (2)
fb800x600                           Use fixed framebuffer graphics (3)
```

```
knoppix keyboard=us xkeyboard=us    Use different keyboard (text/X)
knoppix splash                      Boot with fancy background splashscreen
knoppix toram                       Copy CD to RAM and run from there
knoppix tohd=/dev/hda1              Copy CD to HD partition and run from there
knoppix fromhd=/dev/hda1            Boot from previously copied CD-Image
knoppix testcd                      Check CD data integrity and md5sums
expert                              Interactive setup for experts
```

Hint: Using the default DE-bootimage, SYSLINUX boots with german keyboard layout. The '=' letter is located at Shift-0 on this keyboard (just in case you want to change keyboard and language with lang=us).

*) Try "knoppix pci=irqmask=0x0e98" if (you have a notebook and) your PS/2 mouse doesn't work. (Possibly caused by a BIOS-flaw on your board, BIOS updates can help.) Sometimes, switching to the text console with Control-Alt-F1 and back to the X-screen with Control-F5 solves the problem without rebooting, since the X server reinitializes the mouse driver during that procedure.

If your KNOPPIX CD makes strange noises during boot, or you see frequent errors like "cloop: read error", or programs on your KDE desktop keep crashing randomly, then your CD image is probably defective or incomplete, or your CD-burner created a defective CD due to wrong writing speed or bad media. This is the most common error reported. Please boot with "knoppix testcd" to check if the CD is OK, and/or even better, verify the MD5 checksums that are present on the mirrors before writing the CD. In some cases, defective IDE controllers cause this error if you have DMA enabled. Also, please read the KNOPPIX-FAQ.

In case of a failing hardware autodetection, try booting with any of the "no-" options as shown in the table above, like in
   knoppix noagp noaudio noddc noapm noapic nodma nopcmcia noscsi nousb
to skip some critical parts of the autodetection system.

The "noswap" option is useful for a forensic analysis without touching existing swap partitions.

Some Boards apparently don't pass the proper memory size to the linux-kernel. It may cause the message "Panic: cannot mount root file system" and the system hangs. Use "knoppix mem=128M" to solve that problem if your system has 128MByte memory for example (caution: you MUST use a capital "M" here).

The "expert" mode provides a very simple interface to loading additional Kernel modules from floppy disks (ext2 or vfat), plus interactive configuration of mouse/keyboard/soundcard/xserver. "expert" mode supports

the same boot options as "knoppix".

The "floppyconfig" or "(my)config=/dev/partition" options allow you to reconfigure the system after autoconfiguration by running a bourne shell script called "knoppix.sh" from the root directory on the given device (or floppy). There is a GUI to create such a configuration floppy disk calles "saveconfig" (also located in the KDE menu under "KNOPPIX", but experts also know how to do this by creating their own shellscripts. From Version 2.1 and up, a file called "knoppix.sh", if located in the toplevel KNOPPIX directory on CD, will also be executed at startup. This makes ist easier to create customized versions without having to change anything on the compressed filesystem KNOPPIX/KNOPPIX.

SCSI-Emulation is active for all CD-Roms, so IDE CD-Writers should work with the installed versions of cdrecord and cdrdao (or the graphical frontends thereof, k3b for example).

If your BIOS does not support el torito booting from CD, you can create a bootable floppy disk by issuing (on Linux)
dd if=/cdrom/KNOPPIX/boot.img of=/dev/fd0 bs=18k
or, in case of DOS, use the rawrite.exe program provided in the KNOPPIX directory on CD.

If you wish to remaster the CD, please don't forget to specify
-b KNOPPIX/boot.img
for the german version of the bootfloppy, or
-b KNOPPIX/boot-en.img
for the english version, as option to mkisofs. Otherwise your CD won't be bootable. The directory KNOPPIX, containig the compressed filesystem file "KNOPPIX", must be located in the top level directory of the CD.

Caution: X-Screensaver: Don't start xlock or any screensaver that requires a password. There are no default passwords on KNOPPIX, i.e. all accounts are LOCKED unless you explicitly set a password. See also README_Security.txt about this issue.
If you accidentially hit the screensaver button in KDE, switch to one of the textconsoles by Control-Alt-F1 and kill the screensaver (or just set a password for the knoppix user).

If you would like to edit your X-Server configuration manually (/etc/X11/XF86Config-4 for XFree86 V4.x), use "knoppix 2" to boot into runlevel 2 (textmode only) and, after changing the X configuration, start the X environment with "init 5". Note that you can always leave the graphical environment with "init 2", and restart it later with "init 5".

References

[1] Jones, Pamela, *Groklaw Timeline Project*,
"http://www.groklaw.com/article.php?story=20040203213604869"

[2] Knoppix Cheat Codes,
"ftp://csociety-ftp.ecn.purdue.edu/pub/knoppix/knoppix-cheatcodes.txt"

[3] Gnu-Arch wiki,
"http://wiki.gnuarch.org"

[4] Gnu-Arch IRC channel,
"#arch on irc.freenode.net"

[5] Blackwell, James,
"http://www.inframix.com"

[6] Kernel Traffic,
"http://kt.zork.net/kernel-traffic/latest.html"

[7] Maxwell, Scott, *Linux Core Kernel Commentary*,
CoriolisOpen Press, Scottsdale, AZ, (1999), ISBN 1-57610-469-9

[8] Kernel Maintainers,
"http://kernelnewbies.org/maintainers"

[9] County Energy, *"County Energy bets business on open source"*
"http://www.computerworld.com.au/index.php?id=1025456490&fp=16&fpid=0"

[10] Eben Moglen and OSDL,
"http://www.osdl.org/docs/osdl_eben_moglen_third_statement.pdf"

[11] OSDL quote,
"http://www.groklaw.net/article.php?story=20040210080458683"

[12] Groklaw IRC Channel,
"#groklaw on irc.fdfnet.net"

[13] User Mode Linux,
"http://user-mode-linux.sourceforge.net"

[14] Japan, South Korea and China,
"Japan, South Korea and China discuss working to promote Linux",
"http://seattlepi.nwsource.com/business/140512_asianwindows20.html"

[15] The Register,
"MS in Peruvian open-source nightmare",
"http://www.theregister.co.uk/content/4/25157.html"

[16] GIF Patent,
Why There Are No GIF files on GNU Web Pages",
"http://www.gnu.org/philosophy/gif.html"

[17] Geer, D., Pfleeger, C., Schneier, B. Quarterman, J., Metzger, P. Bace, R., Gutmann, P.
"CyberInsecurity Report"
"http://www.ccianet.org/papers/cyberinsecurity.pdf"