

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRUNG TÂM TÍNH TOÁN
HIỆU NĂNG CAO

KHOA CÔNG NGHỆ
THÔNG TIN

NGUYỄN THANH THUỶ (CHỦ BIÊN)
NGUYỄN QUANG HUY
NGUYỄN HỮU ĐỨC
ĐINH LAN ANH

Nhập môn hệ điều hành

Linux



NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRUNG TÂM TÍNH TOÁN KHOA CÔNG NGHỆ THÔNG TIN
HIỆU NĂNG CAO

NGUYỄN THANH THUỶ (Chủ biên)
NGUYỄN QUANG HUY, NGUYỄN HỮU ĐỨC, ĐINH LAN ANH

NHẬP MÔN HỆ ĐIỀU HÀNH LINUX

(In lần thứ 2, có bổ sung và sửa chữa)



NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

HÀ NỘI - 2005

Lời nói đầu

Trong chương trình đào tạo kỹ sư, cử nhân chuyên ngành về Công nghệ Thông tin (CNTT) và Tin học các kiến thức cơ sở về hệ điều hành có vai trò rất quan trọng. Để có những hiểu biết kỹ lưỡng các đặc trưng cơ bản và các nguyên lý xây dựng hệ điều hành cho máy tính điện tử, các sinh viên CNTT phải tìm tòi sáng tạo những bí ẩn bên trong các hệ điều hành đã có trên thị trường. Bên cạnh các hệ điều hành truyền thống như Windows/WindowNT và UNIX, Linux được đánh giá là một hệ điều hành có nhiều triển vọng.

Linux là một hệ điều hành họ UNIX miễn phí dùng cho máy tính cá nhân đang được sử dụng rộng rãi hiện nay. Được viết vào năm 1991 bởi Linus Torvalds, hệ điều hành Linux đã thu được những thành công nhất định. Hiện nay, Linux ngày càng phát triển, được đánh giá cao và thu hút nhiều sự quan tâm của các nhà tin học.

Là một hệ điều hành đa nhiệm, đa người dùng, Linux có thể chạy được trên nhiều nền phần cứng khác nhau. Linux hỗ trợ chuẩn POSIX, tức là hoàn toàn tương thích với các hệ UNIX khác. Với tính năng ổn định và mềm dẻo, Linux đang dần được sử dụng nhiều trên các máy chủ cũng như máy trạm trong các mạng máy tính. Linux còn cho phép dễ dàng thực hiện việc tích hợp nó và các hệ điều hành khác trong một mạng máy tính như Windows, Novell, Apple... Ngoài ra, với tính năng mã nguồn mở, hệ điều hành này còn cho phép khả năng tùy biến cao, thích hợp cho các nhu cầu sử dụng cụ thể.

Cũng như các hệ Unix khác, việc quản trị hệ thống Linux nói chung là một việc tốn khá nhiều công sức. Công việc này không những đòi hỏi sự hiểu biết về hệ thống, về mạng máy tính mà còn đòi hỏi đức tính cẩn thận, kiên trì, đầu óc tổ chức của người quản trị.

Trong những năm gần đây, hệ điều hành Linux từng bước đã được đưa vào sử dụng tại Việt Nam. Nhiều tổ chức, công ty và các dự án tin học đã chọn Linux là môi trường để phát triển các ứng dụng của mình. Chính vì thế nhu cầu tìm hiểu hệ điều hành này đang trở nên rất lớn. Để đáp ứng yêu cầu nêu trên, chúng tôi cho

ra mắt bộ sách gồm 3 cuốn : “**Nhập môn hệ điều hành Linux**”, “**Quản trị hệ thống Linux**” và “**Bên trong nhân hệ điều hành Linux**”. Bộ sách này được biên soạn dựa trên những kinh nghiệm tích lũy ban đầu của chúng tôi trong quá trình thực hiện dự án “**Xây dựng hệ thống tính toán hiệu năng cao**” (trong khuôn khổ **Chương trình Quốc gia về CNTT**, nay là **Chương trình Kinh tế Kỹ thuật về CNTT**), trong đó hệ điều hành Linux được sử dụng như hệ điều hành mạng với nhiều máy tính được kết chùm với nhau nhằm phục vụ cho việc giải quyết các bài toán cỡ lớn.

Cuốn sách “**Nhập môn hệ điều hành Linux**” này được chia làm 6 chương và 4 phụ lục:

Chương 1. Giới thiệu về hệ điều hành Linux.

Chương này bao gồm lịch sử vắn tắt của hệ điều hành Linux, lý do dùng hệ điều hành Linux, các đặc điểm của hệ thống và một số so sánh giữa Linux và hệ điều hành cổ điển DOS.

Chương 2. Làm quen với Linux : Các lệnh cơ bản.

Chương 2 giúp người sử dụng làm quen với môi trường hệ điều hành Linux, các khái niệm cơ bản của Linux, các lệnh cơ bản bắt đầu từ việc đăng nhập và thoát khỏi hệ thống đến các lệnh cơ bản về tệp và thư mục.

Chương 3. Hệ thống XWindows.

Chương này giới thiệu về hệ thống giao diện XWindows của hệ điều hành Linux: các khái niệm chung, quá trình khởi động XWindows và giới thiệu chi tiết cho người sử dụng môi trường GNOME thông dụng.

Chương 4. Cơ bản về quản trị hệ thống.

Chương này dành cho những người sử dụng chuyên nghiệp hoặc các quản trị viên hệ thống. Nội dung của chương trình bày các thao tác cơ bản trong quản trị hệ thống như: quản trị hệ thống file, quản lý người dùng, sao lưu hệ thống, thay đổi cấu hình hạt nhân hệ điều hành và các thao tác trong trường hợp khẩn cấp.

Chương 5. Sử dụng mạng trong Linux.

Chương 5 đề cập đến vấn đề sử dụng mạng trong hệ điều hành Linux: các khái niệm cơ bản về mạng, các thiết lập cấu hình mạng TCP/IP trong Linux và các sử dụng Linux để kết nối với mạng thông tin toàn cầu Internet.

Chương 6. Các ứng dụng và tiện ích trên Linux.

Chương này giới thiệu các ứng dụng và tiện ích thông dụng trên Linux như các hệ soạn thảo vi, Emacs, tiện ích mc, các khái niệm về Web, email, cách sử dụng trình duyệt thông dụng trên Linux Netscape navigator.

Cuối cùng là các phụ lục hữu ích đối với người sử dụng:

Phụ lục 1: Hướng dẫn cài đặt hệ điều hành Linux RedHat 6.1

Phần này hướng dẫn người sử dụng tự cài đặt hệ điều hành Linux trên máy tính của mình.

Phụ lục 2: Giới thiệu về các ngôn ngữ lập trình trên Linux.

Phần này dành cho những người sử dụng quan tâm đến việc phát triển phần mềm trên Linux. Phụ lục giới thiệu chung về các ngôn ngữ được hỗ trợ trên Linux và địa chỉ để người sử dụng có thể tải về miễn phí.

Phụ lục 3: Các nguồn thông tin về Linux

Đây là phần rất bổ ích cho những ai muốn tìm hiểu thêm về hệ điều hành Linux, chỉ rõ các tài liệu và các địa chỉ Internet liên quan đến Linux.

Phụ lục 4: Các câu hỏi thường gặp (FAQs)

Phần này gồm các câu hỏi thường gặp nhất khi mới sử dụng hệ điều hành Linux. Bạn đọc có thể tìm thấy ở đây những thông tin bổ ích.

Trong quá trình biên soạn, chúng tôi đã hết sức cố gắng, song không tránh khỏi những sai sót. Rất mong nhận được những ý kiến đóng góp và nhận xét của độc giả để có thể hoàn thiện cuốn sách trong những lần xuất bản tiếp theo. Mọi ý kiến đóng góp xin gửi về:

Nguyễn Thanh Thủy

Khoa Công nghệ Thông tin, Trường Đại học Bách khoa Hà nội

Tel 8 696 124, email: thuynt@it-hut.edu.vn

Để cuốn sách sớm ra mắt bạn đọc, chúng tôi đã nhận được nhiều động viên quý báu của Ban chủ nhiệm **Chương trình Kinh tế Kỹ thuật về CNTT**, Ban Giám hiệu trường Đại học Bách khoa Hà nội, và những đóng góp về chuyên môn của các Giáo sư, Thầy cô giáo trong Khoa Công nghệ Thông tin. Chúng tôi xin chân thành cảm ơn những sự giúp đỡ hiệu quả đó.

Các tác giả

Hà nội, Thu 2000

Lời tựa cho lần tái bản thứ nhất

Kể từ lần xuất bản đầu tiên năm 2000 đến nay, cuốn sách của chúng tôi luôn nhận được sự quan tâm, nhiệt tình đóng góp của đông đảo các Thầy Cô, ... Chuyên gia, cũng như các bạn sinh viên tại các khoa Công nghệ Thông tin, các khoa chuyên môn có liên quan tại các trường đại học, cao đẳng trong cả nước. Chúng tôi xin trân trọng cảm ơn sự động viên quý báu của bạn đọc đông đảo gần xa.

Có một thực tế là các hệ điều hành (HĐH) LINUX được cung cấp bởi nhiều distro, do đó HĐH LINUX có những tên gọi khác nhau như RedHat, Fedora, Mandrake, Suse hay Debian. Điểm chung của tất cả HĐH này là phần nhân (phần lõi) của HĐH LINUX. Sự khác biệt nằm ở các tiện ích, các phần mềm đi cùng. Tùy thuộc vào các yêu cầu của người dùng, các distro sẽ đóng gói phần nhân LINUX với các tiện ích khác nhau, do đó việc cài đặt, cấu hình cũng có thể không đồng nhất. Phụ lục 5 mà chúng tôi đưa vào cuốn sách trong lần tái bản có chỉnh lý và bổ sung này chính là nhằm hỗ trợ bạn đọc trong việc tìm hiểu thêm. Hy vọng giúp ích bạn đọc phần nào trong công việc và học tập.

Trong những năm gần đây, cùng với xu thế tìm hiểu và phát triển các ứng dụng theo tiếp cận nguồn mở, trong đó HĐH LINUX được coi là một sản phẩm điển hình, nhiều tổ chức, công ty và các dự án tin học đã chọn Linux là môi trường để phát triển các ứng dụng của mình. Nhiều khoa Công nghệ thông tin và khoa chuyên môn có liên quan đến công nghệ thông tin đã chính thức đưa việc giảng dạy HĐH LINUX vào chương trình đào tạo. Chúng tôi hy vọng sẽ sớm giới thiệu với bạn đọc các cuốn sách "Bên trong nhân hệ điều hành Linux" và "Lập trình LINUX" trong thời gian sớm nhất.

Trong quá trình biên soạn, chúng tôi đã hết sức cố gắng, song không tránh khỏi những sai sót. Rất mong nhận được những ý kiến đóng góp và nhận xét của độc giả để có thể hoàn thiện cuốn sách trong những lần xuất bản tiếp theo. Mọi ý kiến đóng góp xin gửi về:

Nguyễn Thanh Thủy, Khoa Công nghệ Thông tin

Trường Đại học Bách khoa Hà nội

Tel (04)8.696.124, Email: thuynt@it-hut.edu.vn

Chúng tôi xin chân thành cảm ơn sự giúp đỡ hiệu quả của Chương trình Kinh tế Kỹ thuật về CNTT (Bộ KHCN), Ban Giám hiệu trường Đại học Bách khoa Hà nội và những đóng góp về chuyên môn của các Giáo sư, Thầy cô giáo trong Khoa Công nghệ Thông tin.

Các tác giả

Hà nội, xuân 2005

Mục lục

Chương 1. GIỚI THIỆU VỀ HỆ ĐIỀU HÀNH LINUX.....	9
1.1 TỔ CHỨC CỦA CUỐN SÁCH.....	10
1.2 LỊCH SỬ HỆ ĐIỀU HÀNH LINUX	10
1.3 TẠI SAO DÙNG LINUX ?	11
1.4 CÁC ĐẶC ĐIỂM CỦA HỆ THỐNG.....	12
1.5 CÁC ĐẶC ĐIỂM PHẦN MỀM	14
1.6 SO SÁNH LINUX VÀ DOS	16
1.6.1 Các tiến trình.....	16
1.6.2 Các kênh vào ra chuẩn <i>STDIN</i> , <i>STDOUT</i> , và <i>STDERR</i>	16
1.6.3 Các tuyến dẫn và tuyến điều hướng.....	17
1.6.4 Các file thi hành được nối kết động.....	17
1.6.5 Cấu trúc ổ đĩa.....	18
1.6.6 Cấu trúc thư mục	19
1.6.7 Các tệp đặc biệt.....	20
1.6.8 Shell.....	20
1.6.9 Hệ thống cửa sổ X.....	20
1.6.10 TCP/IP và mạng	20
Chương 2 LÀM QUEN VỚI LINUX, CÁC LỆNH CƠ BẢN	21
2.1 LÀM QUEN VỚI LINUX.....	21
2.1.1 Bắt đầu phiên làm việc	21
2.1.2 Kết thúc phiên làm việc	22
2.1.3 Thay đổi mật khẩu	22
2.1.4 Các console ảo	22
2.2 CÁC LỆNH CƠ BẢN	22
2.2.1 Cách dùng lệnh của Linux.....	22
2.2.2 Các lệnh về thư mục	24
2.2.3 Các lệnh về tệp	27
2.2.4 Các lệnh khác	31
2.3 CÁC KHÁI NIỆM CƠ BẢN.....	33
2.3.1 Shell	33
2.3.2 Sự tự động hoàn thành lệnh	33
2.4 TÊN FILE VÀ TÊN THƯ MỤC.....	34

2.5	CHUYỂN HƯỚNG ĐẦU RA.....	35
2.5.1	Chuyển hướng đầu vào	35
2.5.2	Các lệnh của Linux.....	36
2.5.3	Các lệnh được thực hiện nền	37
2.5.4	Giúp đỡ bằng lệnh man.....	37
2.5.5	Các thuộc tính của file.....	38
2.5.6	Các file cấu hình thực hiện	44
2.5.7	Các thư mục quan trọng	46
2.5.8	Các tiến trình	47
2.5.9	Các tuyến dẫn (pipes).....	51
2.5.10	Nối các lệnh thành chuỗi.....	51
	Chương 3 HỆ THỐNG X WINDOW.....	53
3.1	KHÁI NIỆM X WINDOWS.....	53
3.2	XINIT	53
3.3	GNOME.....	54
3.3.1	Giới thiệu chung.....	55
3.3.2	Trình quản lí cửa sổ và GNOME.....	56
3.3.3	Trình quản lí cửa sổ 'Enlightenment'	56
3.3.4	Desktop	70
3.3.5	Các Folder.....	73
3.3.6	Trình quản lí File của GNOME	74
3.3.7	Thanh Panel	90
3.3.8	Tìm sự trợ giúp (Finding Help).....	96
	Chương 4 CƠ BẢN VỀ QUẢN TRỊ HỆ THỐNG	98
4.1	KHỞI ĐỘNG VÀ THOÁT KHỎI HỆ THỐNG	99
4.1.1	Khởi động sử dụng đĩa mềm khởi động.....	99
4.1.2	Khởi động sử dụng LILO	100
4.1.3	Thoát khỏi hệ thống	103
4.1.4	File /etc/inittab	104
4.2	QUẢN TRỊ HỆ THỐNG FILE.....	107
4.2.1	Gắn kết (mount) và tháo bỏ (unmount) hệ thống file.....	107
4.2.2	Gắn kết các hệ thống file tự động bằng file /etc/fstab	109
4.2.3	Các kiểu hệ thống file	111
4.2.4	Các giá trị của trường option (hướng dẫn xử lí file).....	112
4.2.5	Quản lí không gian đĩa	113
4.2.6	File liên kết (link)	119
4.3	QUẢN LÝ NGƯỜI DÙNG	120
4.3.1	Tài khoản siêu người dùng (superuser account).....	120

4.3.2	Thiết lập tài khoản người sử dụng	121
4.3.3	Sử dụng nhóm	128
4.3.4	Lệnh su (switch user).....	132
4.4	SAO LƯU VÀ KHÔI PHỤC	133
4.4.1	Tại sao phải sao lưu	133
4.4.2	Chọn thiết bị để sao lưu.....	134
4.4.3	Đặt kế hoạch sao lưu.....	134
4.4.4	Giữ các log của sao lưu	136
4.4.5	Dùng tar để sao lưu.....	136
4.5	THAY ĐỔI CẤU HÌNH HẠT NHÂN	140
4.5.1	Nâng cấp và cài đặt phần mềm hạt nhân mới	140
4.5.2	Dịch lại hạt nhân từ mã nguồn	141
4.5.3	Thêm các trình điều khiển thiết bị vào hạt nhân	143
4.5.4	Nâng cấp thư viện	143
4.5.5	Sử dụng chương trình dịch C	144
4.6	TRONG TRƯỜNG HỢP KHẨN CẤP	146
4.6.1	Khôi phục bằng đĩa bảo vệ.....	146
4.6.2	Sửa mật khẩu root.....	147
4.6.3	Hệ thống file bị rác.....	147
4.6.4	Câu các file đã mất.....	147
Chương 5 SỬ DỤNG MẠNG TRONG LINUX.....		149
5.1	GIỚI THIỆU CHUNG VỀ MẠNG TRONG LINUX.....	149
5.2	THIẾT LẬP CẤU HÌNH MẠNG TCP/IP TRONG LINUX	150
5.2.1	Khái niệm mạng TCP/IP	150
5.2.2	Địa chỉ IP	150
5.2.3	Tên máy (hostname)	153
5.2.4	Bảng chọn đường	153
5.2.5	Một số lệnh cơ bản: rcp, rlogin, telnet,	154
5.2.6	NFS và NIS.....	158
5.3	KẾT NỐI INTERNET.....	159
Chương 6 CÁC ỨNG DỤNG VÀ TIỆN ÍCH TRÊN LINUX.....		162
6.1	CÁC HỆ SOẠN THẢO, ĐỒ HOẠ VÀ IN ẤN	162
6.1.1	Soạn thảo bằng vi	162
6.1.2	Trình soạn thảo Emacs.....	166
6.1.3	Sử dụng ImageMagick	168
6.1.4	In ấn	169
6.2	TIỆN ÍCH MC	169

6.3	INTERNET và WEB/MAIL	170
6.3.1.	<i>Giới thiệu về Web</i>	170
6.3.2.	<i>Thư điện tử (E-mail)</i>	170
6.3.3.	<i>Trình duyệt Web netscape</i>	171
Phụ lục 1.	Hướng dẫn cài đặt RedHat 6.1.....	172
Phụ lục 2.	Các ngôn ngữ lập trình.....	206
Phụ lục 3.	Các nguồn thông tin về Linux.....	208
Phụ lục 4.	Các câu hỏi thường gặp (FAQs).....	209
Phụ lục 5.	Một số nhà cung cấp HĐH Linux phổ biến	215
Tài liệu tham khảo.....		219

Chương 1. GIỚI THIỆU VỀ HỆ ĐIỀU HÀNH LINUX

Cuốn sách này sẽ giúp các bạn làm chủ hệ điều hành Linux — hệ điều hành mô phỏng Unix cho các máy tính cá nhân — Linux hỗ trợ đầy đủ khả năng đa nhiệm, đa người dùng, hệ thống giao diện X Window, mạng TCP/IP và rất nhiều tính năng khác mà bạn sẽ lần lượt được làm quen trong cuốn sách này.

Linux là hệ điều hành thu hút được nhiều sự chú ý nhất trong vòng vài năm trở lại đây. Ngay từ khi xuất hiện, nó đã được lan rộng một cách nhanh chóng và biết tới như một hệ điều hành UNIX-với mã nguồn mở. Thật ngạc nhiên, sự thành công của Linux có được nhờ sự làm lại một trong những hệ điều hành lâu đời nhất và hiện vẫn đang được sử dụng rộng rãi — hệ điều hành UNIX. Linux bao gồm cả các công nghệ cũ và mới.

Nhìn từ góc độ kỹ thuật, Linux chỉ là một nhân hệ điều hành, nó hỗ trợ đầy đủ các phục vụ cơ bản về quản lý tiến trình, bộ nhớ ảo, quản lý file và vào ra thiết bị. Nói cách khác, bản thân Linux là phần thấp nhất của hệ điều hành.

Tuy nhiên, phần lớn người dùng đều coi "Linux" như một hệ thống hoàn chỉnh gồm nhân hệ điều hành kèm theo các trình ứng dụng khác: một môi trường làm việc và phát triển đầy đủ bao gồm trình dịch, các hệ soạn thảo, giao diện đồ họa, xử lý văn bản,... Hiện nay, cho tới phiên bản Linux RedHat 6.1, Linux đã trở thành một hệ điều hành đầy đủ cho thương mại, giáo dục hoặc người dùng cá nhân.

Linux có thể được cài đặt trên một máy tính cá nhân và trở thành một trạm làm việc với đầy đủ sức mạnh của UNIX. Linux cũng có thể được sử dụng với mục đích thương mại trên một mạng máy tính như một môi trường tính toán và truyền tin. Trong các trường đại học, Linux được sử dụng để giảng dạy về hệ điều hành và lập trình hệ điều hành. Và tất nhiên, Linux cũng có thể được sử dụng trên các máy tính cá nhân như các hệ điều hành khác.

Ngoài chức năng trạm làm việc và người sử dụng đơn lẻ, Linux còn được sử dụng trên các máy chủ lớn. Linux ngày càng tỏ ra là một hệ điều hành mạnh, ổn định và đủ mềm dẻo để chạy trên các hệ thống với bộ nhớ lớn và đa xử lý. Linux còn được sử dụng để nối ghép nhiều máy tính với nhau tạo thành các "cluster" rất mạnh để giải quyết các nhu cầu tính toán hiệu năng cao.

Điều làm cho Linux trở nên khác biệt là việc viết mã tự do của UNIX. Việc này do một nhóm phát triển tự nguyện trên mạng Internet, họ trao đổi mã nguồn, phát hiện và sửa lỗi trong một môi trường mở.

1.1 TỔ CHỨC CỦA CUỐN SÁCH

Cuốn sách này nhằm mục đích hướng dẫn cho những người mới làm quen với Linux và muốn nắm được những vấn đề cơ bản nhất của Linux. Cuốn sách được chia thành các phần sau:

- Linux là gì? Mục đích và nguyên lý của hệ điều hành này, và nó có thể làm gì cho bạn.
- Phần giới thiệu tổng quan về hệ Linux, bao gồm các khái niệm và các lệnh quan trọng nhất.
- Hệ thống X Window của Linux.
- Hệ quản lý file của Linux.
- Giới thiệu về quản trị hệ thống trong Linux. Những công việc quan trọng nhất mà một người quản trị hệ thống cần làm.
- Các hệ soạn thảo trong Linux
- Các vấn đề về mạng trong Linux.

Phần phụ lục 1 sẽ đề cập đến các ngôn ngữ lập trình trên Linux với mục đích giới thiệu và phần phụ lục 2 chứa một số tham khảo hữu ích khi bạn muốn tìm hiểu thêm hoặc muốn có thêm thông tin về hệ điều hành Linux.

1.2 LỊCH SỬ HỆ ĐIỀU HÀNH LINUX

Linux bắt nguồn từ một hệ điều hành lớn hơn có tên là Unix. Unix là một trong những hệ điều hành được sử dụng rộng rãi nhất trên thế giới do tính ổn định và khả năng hỗ trợ của nó. Ban đầu nó đã được phát triển như một hệ điều hành đa nhiệm cho các máy mini và các máy lớn (mainframe) trong những năm 70. Cho tới nay nó đã được phát triển trở thành một hệ điều hành phổ dụng trên toàn thế giới, mặc dù với giao diện chưa thân thiện và chưa được chuẩn hoá hoàn toàn.

Linux là phiên bản Unix được cung cấp miễn phí, ban đầu được phát triển bởi Linus Torvald năm 1991 khi còn là một sinh viên của trường đại học Helsinki Phần Lan. Hiện nay, Linus làm việc tại tập đoàn Transmeta và tiếp tục phát triển nhân hệ điều hành Linux (Linux kernel).

Khi Linus tung ra miễn phí phiên bản đầu tiên của Linux (0.02) trên Internet, vô tình đã tạo ra một làn sóng phát triển phần mềm lớn nhất từ trước đến nay trên phạm vi toàn cầu. Hiện nay, Linux được phát triển và bảo trì bởi một nhóm hàng nghìn lập trình viên cộng tác chặt chẽ với nhau qua Internet. Nhiều công ty đã xuất

hiện, cung cấp Linux dưới dạng gói phần mềm dễ cài đặt, hoặc cung cấp các máy tính đã cài đặt sẵn Linux.

Tháng 11 năm 1991, Linus đưa ra bản chính thức đầu tiên của Linux, phiên bản 0.02. Ở phiên bản này, Linus đã có thể chạy *bash* và *gcc* (trình dịch C GNU) nhưng mới chỉ dừng lại ở đó. Hệ thống chưa có các hỗ trợ người dùng và tài liệu hướng dẫn. Các số hiệu phiên bản không ngừng gia tăng cùng với việc bổ sung thêm các tính năng mới. Tháng 3/1994, phiên bản Linux 1.0 ra đời, đây là phiên bản tương đối ổn định. Hiện nay, với phiên bản Linux 2.2.6, bạn có thể làm việc trên môi trường đồ họa với các ứng dụng cao cấp như: các tiện ích đồ họa và nhiều tiện ích khác.

Linux khó có thể thành công được như hiện nay nếu không có các công cụ GNU của Tổ chức phần mềm miễn phí (Free Software Foundation). Trình dịch *gcc* của GNU đã giúp cho việc viết mã của Linux dễ dàng hơn rất nhiều. Thậm chí tổ chức này đã yêu cầu các bản Linux với các tiện ích kèm theo phải gọi là GNU/Linux.

Hệ điều hành Berkley Unix (BSD) cũng đóng một vai trò quan trọng đối với Linux trong việc làm cho hệ điều hành này trở nên phổ biến như hiện nay. Hầu hết các tiện ích đi kèm với Linux được chuyển sang từ BSD, đặc biệt là các công cụ về mạng và các tiện ích.

Hiện nay, Linux là một hệ điều hành Unix đầy đủ và độc lập. Nó có thể chạy X Window, TCP/IP, Emacs, Web, thư điện tử và các phần mềm khác. Hầu hết các phần mềm miễn phí và thương mại đều được chuyển lên Linux. Rất nhiều các nhà phát triển phần mềm đã bắt đầu chuyển sang viết trên Linux. Người ta đã thực hiện các phép đo benchmarks trên các hệ Linux và thấy rằng chúng thực hiện nhanh hơn khi thực hiện trên các trạm làm việc Sun Microsystems và Compaq, thậm chí nhiều khi còn nhanh hơn cả trên Windows 98 và Windows NT. Thật khó có thể hình dung được hệ điều hành Unix "tí hon" này phát triển nhanh như thế nào.

1.3 TẠI SAO DÙNG LINUX ?

Nếu bạn đã có máy tính trong tay, đầu tiên bạn phải có một hệ điều hành cài đặt trên đó bạn mới có thể sử dụng được các chương trình ứng dụng. Hệ điều hành là chương trình điều hành mọi hoạt động trong máy tính của bạn, mọi chương trình ứng dụng khác đều chạy trên nền của hệ điều hành này. Linux là hệ điều hành mã nguồn mở, được cung cấp miễn phí cho người sử dụng. Nó có khả năng đa nhiệm, đa xử lý, hỗ trợ mạng, khả năng tương thích phần cứng và nhiều tính năng khác. Sau đây là các tính năng mà bạn có được từ hệ điều hành Linux :

- **Tính ổn định** : Linux có tính ổn định cao, đây là một trong những ưu điểm của Linux so với các hệ điều hành khác. Tính ổn định ở đây có nghĩa là nó ít bị lỗi khi sử dụng so với hầu hết các hệ điều hành khác. Người sử dụng Linux sẽ không phải lo lắng đến chuyện máy tính của mình bị hiện tượng

"treo cứng" khi đang sử dụng nữa. Thông thường lý do để bạn bắt buộc phải khởi động lại hệ thống là do mất điện, nâng cấp phần cứng hoặc phần mềm.

- ✎ **Tính bảo mật** : Khi làm việc trên Linux người dùng có thể yên tâm hơn về tính bảo mật của hệ điều hành. Linux là hệ điều hành đa nhiệm, đa người dùng, điều này có nghĩa là nhiều người sử dụng có thể vào phiên làm việc của mình trên cùng một máy và tại cùng một thời điểm. Linux cung cấp các mức bảo mật khác nhau cho người sử dụng. Mỗi người sử dụng chỉ làm việc trên một không gian tài nguyên dành riêng, chỉ có người quản trị hệ thống mới có quyền thay đổi trong máy.
- ✎ **Tính hoàn chỉnh** : Bản thân Linux đã được kèm theo các trình tiện ích cần thiết. Tất cả các trình tiện ích mà bạn mong đợi đều có sẵn hoặc ở một dạng tương đương rất giống. Trên Linux, các trình biên dịch như C, C++,... các hạt nhân hay TCP/IP đều được chuẩn hoá.
- ✎ **Tính tương thích** : Linux tương thích hầu như hoàn toàn với một số chuẩn UNIX như IEEE POSIX.1, UNIX System V và BSD UNIX. Trên Linux bạn cũng có thể tìm thấy các trình giả lập của DOS và Windows cho phép bạn có thể chạy các ứng dụng quen thuộc trên DOS và Windows. Linux cũng hỗ trợ hầu hết các phần cứng máy PC.
- ✎ **Hệ điều hành 32-bit đầy đủ** : Ngay từ đầu Linux đã là hệ điều hành 32 bit đầy đủ. Điều đó có nghĩa là bạn không còn phải lo về các giới hạn bộ nhớ, các trình điều khiển EMM hay các bộ nhớ mở rộng, ... khi sử dụng Linux.
- ✎ **Dễ cấu hình** : Bạn không còn phải bận tâm về các giới hạn 640K và tiến hành tối ưu hoá bộ nhớ mỗi lần cài đặt một trình điều khiển mới. Linux cho bạn hầu như toàn quyền điều khiển về cách làm việc của hệ thống.
- ✎ **Khả năng làm việc trên nhiều loại máy**: Cấu hình phần cứng tối thiểu mà Linux cần chỉ là chip 80386, 2MB bộ nhớ, 10-20 MB không gian đĩa để bắt đầu. Khi bạn càng bổ sung phần cứng thì Linux chạy càng nhanh. Linux có khả năng chạy trên nhiều dòng máy khác nhau như Apple Macintosh, Sun, Dec Alpha và Power PC.

1.4 CÁC ĐẶC ĐIỂM CỦA HỆ THỐNG

• Các version khác nhau của Linux:

Một trong những điểm dễ nhầm lẫn của những người mới làm quen với Linux là các version của Linux. Khi bạn bắt đầu tiếp cận với Linux, thông thường bạn tìm các phiên bản Linux trên CD ROM như "Red Hat version 6.1" hay "SuSE Linux version 6.3". Các số gắn với một phiên bản chỉ liên quan tới các lần phân phối, một phiên bản với số hiệu lớn hơn không có nghĩa là phần mềm được cập nhật mới hơn. Thông thường các nhân Linux (Linux kernel) có một số hiệu phiên bản riêng. Tại mỗi một thời điểm có hai hai phiên bản mới nhất — phiên bản ổn định (stable) và phiên bản phát triển (development). Phiên bản ổn định dành cho hầu hết

người dùng, còn phiên bản phát triển thay đổi rất nhanh và được chạy thử bởi các nhà phát triển trên Internet. Phiên bản ổn định thường có số hiệu nhỏ hơn, chẳng hạn 2.2 và 2.3. Mỗi phiên bản kernel có thêm một số hiệu thứ ba mô tả lần cập nhật cuối cùng. Chẳng hạn phiên bản Linux kernel ổn định mới nhất vào tháng 7/99 là 2.2.10 và phiên bản phát triển tương ứng là 2.3.11.

- **Các đặc tính của hệ thống:**

Linux là hệ điều hành hoàn toàn đa nhiệm và đa người dùng. Điều này có nghĩa là tại một thời điểm có thể có nhiều người sử dụng và chạy nhiều chương trình trên cùng một máy tính. Linux cũng hỗ trợ các hệ thống đa xử lý (tối 16 bộ xử lý).

Linux tương thích gần như hoàn toàn với các bản Unix chuẩn như IEEE POSIX.1, System V, và BSD.

Linux có thể được cài đặt cùng với các hệ điều hành khác như Windows 95/98, Windows NT, OS/2, hoặc các phiên bản khác của Linux. Chương trình tải hệ thống Linux (LILO) cho phép bạn chọn hệ điều hành khi khởi động.

Linux có thể chạy trên nhiều cấu trúc CPU khác nhau như Intel, SPARC, Alpha, PowerPC, MIPS và m68k.

Linux hỗ trợ nhiều hệ thống file khác nhau như: hệ thống file mở rộng (ext2fs) dành riêng, MS-DOS file, Windows, OS/2, Apple, ...

Mạng là một trong những điểm mạnh của Linux. Linux cung cấp đầy đủ các dịch vụ giao thức mạng TCP/IP, bao gồm các drive thiết bị cho card Ethernet, PPP và SLIP, PLIP (Parallel Line Internet Protocol) và NFS (Network File System). Hỗ trợ các dịch vụ như FTP, Telnet, NNTP và SMTP (Simple Mail Transfer Protocol). Kernel Linux hỗ trợ bức tường lửa (firewall) cho mạng, bạn có thể đặt cấu hình một máy Linux bất kỳ như một firewall. Chúng ta sẽ nói kỹ hơn về mạng trong Linux ở chương 6, TCP/IP và PPP.

- **Kernel:**

Kernel là phần chính, là trái tim của hệ điều hành. Nó nắm nhiệm vụ điều khiển giao diện giữa chương trình người sử dụng với các thiết bị phần cứng, xếp lịch các tiến trình để có thể thực hiện đa nhiệm, và nhiều tác vụ khác của hệ thống. Kernel không phải là một tiến trình chạy riêng biệt trong hệ thống, mà là một tập các trình đơn nằm trong bộ nhớ, mọi tiến trình đều gọi đến chúng. Ví dụ lệnh *read* của hệ thống sẽ đọc dữ liệu từ một file được mô tả, đối với người lập trình nó giống như một hàm trong C, nhưng thực ra mã của lệnh *read* được chứa trong kernel Linux.

Trên nhiều cấu trúc máy tính, kernel có thể mô phỏng các lệnh dấu phẩy động (PFU) nếu hệ thống không có bộ đồng xử lý toán học. Kernel Linux cũng hỗ trợ các kỹ thuật như: phân trang bộ nhớ, bộ nhớ ảo, cache đĩa, ...

1.5 CÁC ĐẶC ĐIỂM PHẦN MỀM

- **Các lệnh cơ bản và các tiện ích :**

Nói chung tất cả các tiện ích và các lệnh cơ bản của Unix đều được chuyển sang Linux: các lệnh cơ bản như *ls*, *awk*, *tr*, *sed*, *hs*, *more*, và các phần mềm như Perl, Python, Java Development Kit. Các trình soạn thảo văn bản cũng được đưa vào trong Linux như: vi, ex, GNU Emacs,... Chúng ta sẽ nói đến các trình soạn thảo kỹ hơn trong chương 8. Điểm đáng chú ý ở đây là hầu hết các tiện ích cơ bản của hệ điều hành Linux đều là GNU và các phiên bản GNU này được đánh giá tốt hơn so với các bản GNU trong các phiên bản Unix chuẩn của BSD và AT&T.

Một trong những tiện ích quan trọng nhất trong Linux là *shell*. *Shell* là một chương trình cho phép đọc và thực hiện các lệnh của người dùng. Trong *shell* người ta có thể viết các *shell script*, tương tự như các file BAT trong MS-DOS, đó là các tệp chứa các chương trình ngôn ngữ lệnh *shell*. Có nhiều loại *shell* trong Linux, chẳng hạn C Shell, Bash shell (GNU Bourne Again Shell), ksh (Korn shell).

- **Các ngôn ngữ lập trình :**

Linux cung cấp một môi trường lập trình Unix đầy đủ, bao gồm mọi thư viện chuẩn, các công cụ lập trình, trình dịch và gỡ rối. Hai ngôn ngữ lập trình phổ biến nhất là C và C++ được hỗ trợ trong Linux với trình dịch *gcc* của GNU. Bộ Java Development Kit của Sun cũng được đưa vào Linux. Các ngôn ngữ lập trình khác cũng được đưa vào Linux như: Smalltalk, Fortran, Pascal, Lisp, ...

- **Hệ thống X Window :**

Hệ thống X Window là giao diện đồ họa chuẩn cho các máy Unix. Đó là một môi trường mạnh hỗ trợ rất nhiều ứng dụng. Phiên bản X Window trên Linux là *XFree86*. Các ứng dụng chuẩn trên X Window là *xterm* (một bộ mô phỏng đầu cuối dùng cho các ứng dụng ở chế độ text), *xdm* (quản lý việc vào ra hệ thống của người dùng) *xclock* (đồng hồ), *xman* (bộ đọc trang hướng dẫn đồ họa), ... Nếu bạn đã từng sử dụng MS Windows thì bạn sẽ thấy rằng môi trường X Window rất giống môi trường MS Windows.

Giao diện X Window được điều khiển bởi chương trình window manager. Chương trình này có trách nhiệm đặt các cửa sổ, giao diện người dùng cho phép thay đổi kích thước, đặt biểu tượng, di chuyển cửa sổ, đặt kiểu của khung cửa sổ v.v... Giao diện X Window được đảm nhiệm bởi chương trình *XFree86*. Chương trình *XFree86* chứa chương trình window manager chuẩn MIT *twm*, đồng thời các chương trình cao cấp khác như Open look Virtual Window manager (*olvwm*) vẫn có sẵn.

Chương trình *XFree86* cũng chứa cả các thư viện lập trình và các file includes cho các nhà phát triển có thể phát triển các ứng dụng trên X Window (X applications). X Window cũng hỗ trợ *Athena*, *Openlook* và *Xaw3D*. Các font chuẩn, bitmaps và các tài liệu cũng có sẵn trong X Window. X Window cũng hỗ trợ các

công cụ đồ hoạ 3 chiều như: *PEX* và *Mesa* (một phiên bản cài đặt miễn phí của *Open GL 3D*).

Trong chương 3 chúng ta sẽ có dịp trở lại với X Window.

- **KDE và GNOME:**

KDE và GNOME là hai dự án quan trọng trong thế giới Linux. Cả hai hệ thống này đều cung cấp một môi trường hoàn toàn tích hợp đồ hoạ chạy trên đỉnh của X Window. Cả hai đều cung cấp đầy đủ window manager, các tiện ích và các ứng dụng vượt trên cả các hệ thống như Windows 98.

Hầu hết các phiên bản Linux đều cho phép đặt cấu hình một cách tự động một trong hai môi trường trên khi cài đặt.

Mặc dù cả hai môi trường KDE và GNOME đều có mục đích chung là làm cho giao diện người sử dụng của Linux trở nên thân thiện hơn, chúng có những mục đích khác nhau. Mục tiêu chính của KDE là dễ sử dụng, ổn định và giao diện người dùng tương thích với các môi trường khác (chẳng hạn Windows 95); trong khi GNOME lại chú ý đến giao diện đẹp mắt và có khả năng đặt cấu hình tối đa.

- **Linux và mạng:**

Linux là một trong những hệ điều hành mạng mạnh nhất, ngày càng có nhiều người chọn Linux làm hệ điều hành mạng. Linux hỗ trợ hai giao thức cơ bản cho các hệ thống Unix: TCP/IP và UUCP.

Hầu hết các mạng TCP/IP đều sử dụng card mạng Ethernet để kết nối. Linux hỗ trợ rất nhiều loại card Ethernet thông dụng cũng như các loại Fast Ethernet, Gigabit Ethernet, ATM, ISDN, mạng LAN không dây, Token Ring, packet radio và các giao diện mạng hiệu năng cao khác.

Linux cũng hỗ trợ PPP và SLIP, cho phép bạn kết nối Internet qua modem. Linux hỗ trợ các trình duyệt Web như Netscape và các web server như *Apache*.

Samba là gói phần mềm cho phép các máy Linux hoạt động như các file server và print server trên Windows. NFS cho phép hệ thống có thể chia sẻ các tệp giữa các máy tính với nhau trên mạng. Với NFS, bạn nhìn các tệp ở xa giống như là trên chính máy tính của bạn. Giao thức FTP (File Transfer Protocol) cho phép truyền các tệp giữa các máy tính trên mạng với nhau.

Các dịch vụ truyền thư điện tử như: sendmail, exim và smail; các dịch vụ telnet, rlogin, ssh, và rsh cho phép bạn có thể truy nhập và làm việc trên một máy tính khác trên mạng. Linux cũng hỗ trợ TCP/IP và cung cấp một giao diện lập trình socket chuẩn.

- **Giao tiếp với Windows và MS-DOS:**

Trong Linux có rất nhiều các tiện ích cho phép bạn có thể giao tiếp với Windows và MS-DOS. Trong đó phải kể đến *Wine*-trình giả lập Microsoft Windows trên X Window trong Linux. Chương trình này cho phép các ứng dụng

trên Windows có thể chạy được trên Linux. Một ứng dụng khác là trình giả lập MS-DOS trên Linux cho phép bạn chạy các ứng dụng dưới DOS trên Linux.

Linux cũng cung cấp một giao diện để chuyển đổi giữa các file giữa Linux và Windows. Bạn có thể đưa một bảng phân hoạch đĩa của Windows trên Linux và có thể truy nhập trực tiếp vào các file của Windows như một file bình thường. Thêm vào đó, Linux có các công cụ như *mttools* cho phép truy nhập trực tiếp vào các đĩa mềm ở khuôn dạng MS-DOS và *htools* cho các đĩa mềm của Macintosh.

1.6 SO SÁNH LINUX VÀ DOS

1.6.1 Các tiến trình

Trong DOS, khi một chương trình được nạp vào bộ nhớ, việc giải phóng nó ra khỏi bộ nhớ thường rất khó khăn. Sở dĩ như vậy là do cách xếp chồng các chương trình trong bộ nhớ của DOS. Nếu bạn nạp vào các chương trình A, B rồi đến C, DOS sẽ đặt các chương trình đó theo thứ tự C, B và A nằm dưới cùng. Để gỡ bỏ chương trình A, ta cũng phải gỡ bỏ các chương trình B và C. Linux khắc phục nhược điểm này bằng cách sao cho khi gỡ bỏ chương trình A sẽ không ảnh hưởng đến B và C. Linux còn cung cấp khả năng bổ sung để dễ dàng điều khiển hoạt động của các tiến trình đang chạy. MS-DOS cũng đưa ra một giải pháp tương tự. Các ứng dụng trên MS-DOS không bắt buộc phải đóng theo thứ tự chúng đã được mở.

Ngoài ra, Linux còn có khả năng tạo ra các vùng trao đổi (swap space). Vùng trao đổi là một phần đĩa cứng được dùng làm bộ nhớ ảo. Các tiến trình không hoạt động hoặc đang đợi có thể được trao đổi trong một quãng thời gian ngắn bằng cách phân trang (paging). Tiến trình này được xóa khỏi bộ nhớ RAM rồi được lưu trữ trong bộ nhớ ảo, giải phóng không gian cho một tiến trình khác. Khi tiến trình này cần sử dụng lại CPU, nó sẽ được trao đổi trở lại RAM nơi nó sẽ bắt đầu lại tiến trình. Tuy nhiên, nếu có quá nhiều phân trang cùng một lúc có thể gây đổ vỡ, ở đó CPU đang dành toàn bộ thời gian của nó để trao đổi bộ nhớ với ổ đĩa cứng, mà không còn thời gian dành cho các tiến trình. Linux có khả năng để ngăn ngừa sự đổ vỡ, song nó vẫn có thể xảy ra. Để ngăn ngừa sự đổ vỡ, bạn phải đảm bảo có đủ bộ nhớ RAM. Với một hệ thống không chạy X (một trong các hệ thống lập cửa sổ của Linux), 8 MB RAM thường là đủ với khoảng 16 MB cho vùng trao đổi. Với các hệ thống chạy X, 16 MB RAM và 16-32MB cho vùng trao đổi là thỏa đáng.

1.6.2 Các kênh vào ra chuẩn STDIN, STDOUT, và STDERR

Khác với DOS, Linux đưa ra khái niệm các kênh vào ra chuẩn STDIN, STDOUT, và STDERR. STDIN (*standard input*) là kênh vào chuẩn của một chương trình. STDOUT là kênh xuất chuẩn của các chương trình. STDERR là kênh lỗi chuẩn, đây là cách kết xuất đặc biệt sao cho chương trình không bị kẹt trong một tuyến dẫn (pipe) hoặc tuyến điều hướng (redirection) (xem bên dưới) trong các hoàn cảnh bình thường. Theo cách này, nếu gặp phải một lỗi bất ngờ song vô hại từ

một chương trình, nó sẽ không làm hỏng dữ liệu mà chương trình kế tiếp đang nhận.

1.6.3 Các tuyến dẫn và tuyến điều hướng

Các tuyến dẫn và tuyến điều hướng tồn tại trong DOS. Các tuyến dẫn cho phép trực tiếp gửi kết xuất của một chương trình sang thành đầu vào của một chương trình khác. Tuyến điều hướng cho phép sử dụng một tệp thay cho STDIN, STDOUT, hoặc STDERR. Ví dụ, bạn có thể dùng một tuyến dẫn hoặc tuyến điều hướng nếu chạy một trong hai lệnh dưới đây:

```
more < file.txt
type file.txt | more
```

Các lệnh trên đây cho ra cùng kết xuất, song bằng hai biện pháp khác nhau. Trong cách thứ nhất, tệp có tên file.txt đã được dùng làm đầu vào chuẩn cho lệnh more thay vì được gõ từ bàn phím. Phương pháp thứ hai sử dụng lệnh type để đọc vào tệp file.txt, và gửi nó ra khỏi STDOUT để STDIN của lệnh more tiếp nhận.

1.6.4 Các file thi hành được nối kết động

Trên hệ điều hành DOS, các chương trình được biên dịch và liên kết với tất cả các hàm bên trong thành một tệp thi hành duy nhất gọi là tệp *kết nối tĩnh*. Các thư viện của tệp kết nối tĩnh và sẽ không thay đổi trừ phi bạn biên dịch lại chương trình. Hệ điều hành MS Windows đưa ra một ý tưởng tương tự như trong Linux: Dynamically Linked Library (thư viện kết nối động). Có nghĩa là bạn có thể tạo một tệp thi hành nhỏ hơn bằng cách lưu trữ các hàm C chung (như printf()) tại một vị trí tập trung nơi nhiều tiến trình có thể dùng hàm đó. Điều này có một số ưu điểm:

1. Các thư viện của chương trình có thể thay đổi dễ dàng. Nếu tiến trình tạo mã bên trong của một hàm như printf() thay đổi, ta không cần biên dịch lại toàn bộ phần mềm. Chỉ việc cài đặt các hàm thư viện mới và tiếp tục chạy các chương trình của bạn.
2. Các tệp thi hành chiếm ít chỗ hơn. Nếu bạn có 1000 chương trình mà mỗi chương trình dành 5K đáng giá cho các hàm C, tổng cộng sẽ lên tới 5MB với 1000 tệp. Tuy nhiên, nếu có một vị trí cho tất cả các hàm C chung, nó có thể chiếm 600K, chứa thêm 5K cho các hàm. Dùng vị trí tập trung có thể giúp ta tiết kiệm được trên 4MB trong tình huống này.

Hệ thống nối kết động này cũng có các nhược điểm như:

1. Các tệp thi hành khởi động chậm hơn. Do hệ lõi phải nạp thêm các tệp để khởi động chương trình, một chương trình nối kết động sẽ nạp lâu hơn so với chương trình nối kết tĩnh.
2. Nếu vô tình, bạn xóa các thư viện mà Linux sử dụng, nhiều chương trình sẽ không làm việc. Nhiều chương trình dựa vào các thư viện để làm việc

đúng đắn. Để giữ cho hệ thống Linux khả dụng trong tình huống này, vài chương trình cần thiết chạy Linux đã được nối kết tĩnh, để bạn có thể đưa các thư viện trở lại. Chúng thường nằm trong /sbin hoặc /usr/sbin.

3. Tốt hơn ta nên có các thư viện. Có những trường hợp một chương trình cần sử dụng các thư viện mà có thể bạn chưa cài đặt. Nếu không có các thư viện, bạn sẽ phải tìm chúng hoặc dùng một phiên bản nối kết tĩnh của chương trình. Điều này không xảy ra thường xuyên, bởi hầu hết các thư viện mà bạn cần đều có trong bản phân phối Linux.

Một dạng thức mới để lưu trữ các thư viện đang nhanh chóng trở thành phổ dụng, có tên là ELF (Executable Linking Format = dạng thức nối kết tệp thi hành), và đưa ra một cách hơi thông minh hơn để tạo các thư viện dùng chung. Trong dạng thức cũ (có tên a.out), các thư viện phải nằm trong các phân cụ thể của bộ nhớ. Để ngăn ngừa hai thư viện lọt vào cùng một vị trí trong bộ nhớ, một bảng đã được xây dựng và mỗi nhà phát triển thư viện đều phải đăng ký thư viện của họ. Ngoài ra, việc tạo các thư viện dùng chung thường khó thực hiện hơn, bởi các nhà phát triển cần biết các vị trí bộ nhớ và các thông tin khác để đặt các thư viện của họ đúng đắn. Dạng thức ELF cho phép đặt các thư viện tại bất kỳ đâu trong bộ nhớ, nó sẽ chăm sóc cả bảng lẫn việc lập trình bộ nhớ. Mặt yếu đó là các chương trình phải được biên dịch lại để sử dụng ELF. Các thư viện a.out và ELF có thể nằm trong bộ nhớ cùng một lúc, nhưng hai thư viện không tương thích trực tiếp.

1.6.5 Cấu trúc ổ đĩa

Linux lưu trữ các tệp không theo dạng có các ổ đĩa A: hoặc C: như trong DOS hoặc MS Windows.

Với MS-DOS, trước tiên bạn phân hoạch ổ đĩa bằng một chương trình như FDISK để báo cho ổ đĩa biết nó sẽ được tách nhỏ như thế nào. Sau đó, bạn dùng lệnh FORMAT để xác lập phân hoạch đó sẽ dùng trong DOS. Và sau đó nó được gọi là một ổ đĩa, như ổ đĩa C hoặc E vân vân.

Trong Linux, bạn vẫn phân hoạch ổ đĩa vật lý để tách riêng nó. Sau đó, dùng một chương trình như mke2fs để định dạng phân hoạch, mà giờ đây được gọi là một hệ thống tệp (filesystem).

Người dùng chỉ làm việc với một ổ đĩa, và mọi thứ được nêu theo dạng các thư mục con của ổ đĩa đó trước người dùng. Do phải có một thư mục gốc, một thư mục nơi ráp nối mọi thứ từ đó, nên Linux cung cấp một thư mục gốc, tên /.

Bạn có thể có một hệ tệp cho thư mục gốc và các tệp khác (thư mục /), một hệ thống tệp khác cho các lệnh người dùng (thư mục /usr) và một hệ thống tệp khác cho không gian người dùng (thư mục /home hoặc user).

Khi Linux khởi tạo, nó dựng lên (mount) thư mục gốc, sau đó tất cả các hệ thống tệp khác đều được dựng lên từ thư mục gốc.

Có vài ưu điểm khi dùng hệ thống này. Trước hết, bạn có thể bổ sung thêm tổng không gian vào hệ thống bằng cách bổ sung các phân hoạch. Nếu lưu ý thấy

rằng thư mục con `/usr/x11r6` sẽ choán nhiều chỗ hơn `/usr`, bạn có thể tạo một hệ thống tệp mới. Có thể tạo hệ thống tệp mới này từ phần còn rảnh của một ổ đĩa, hoặc từ một ổ đĩa mới. Sau đó có thể ráp nối phân hoạch mới làm `/usr/x11r6`. Thứ hai, hầu hết các thay đổi trong phân hoạch đều vô hình đối với cả người dùng lẫn các chương trình của người dùng. DOS cần một tên ổ đĩa và một thư mục. Linux chỉ cần một thư mục, và, trong nhiều trường hợp, chương trình sẽ không để ý việc bạn có dùng một tệp trên một phân hoạch khác hay không.

Linux hiểu các ổ đĩa vật lý, như ổ đĩa mềm 3.5", là những tệp đặc biệt nằm trong thư mục `/dev`. Ví dụ, ổ đĩa đầu tiên (có tên `A:` trong DOS) sẽ có tên `/dev/fd0` trong Linux. `Fd` ám chỉ sự việc đó là một ổ đĩa mềm, và `0` nghĩa nó là thiết bị đầu tiên thuộc kiểu đó.

Các ổ đĩa cứng được hiểu theo hai cách khác nhau: chính ổ đĩa vật lý và một phân hoạch trên ổ đĩa vật lý đó. Các mục ổ đĩa cứng trong `/dev` là các tiếp đầu ngữ có hd để chỉ ổ đĩa cứng. Theo sau nó là một 'a' cho ổ đĩa vật lý đầu tiên, một 'b' cho ổ đĩa vật lý thứ hai, vân vân. Nên nhớ, điều này có nghĩa là ổ đĩa vật lý. Sau đó là một con số chỉ định phân hoạch bên trong ổ đĩa vật lý đó. Phân hoạch đầu tiên là `1`, phân hoạch thứ hai là `2`, v.v..

1.6.6 Cấu trúc thư mục

Để các chương trình làm việc được với nhau, Linux đã xác lập một cấu trúc thư mục mà hầu hết mọi người dùng phải tuân theo. Dự án FSSTND (File System Standard-chuẩn cho hệ thống file) là dự án về cách xác lập chuẩn của các thư mục. Sau đây là một số thư mục được đề nghị của dự án:

<code>/etc</code>	Cấu hình hệ thống
<code>/usr/bin</code>	Chứa các lệnh của người dùng
<code>/dev</code>	Các tệp thiết bị
<code>/home</code>	Các thư mục người sử dụng
<code>/usr/local</code>	Các chương trình phụ trội không thuộc thành phần của một hệ thống thông thường. Trong đó có các thư mục bổ sung: <code>/usr/local/bin</code> <code>/usr/local/man</code> <code>/usr/local/lib</code> <code>/usr/local/include</code>
<code>/usr/src</code>	Vị trí mã nguồn (kể cả mã nguồn Linux)
<code>/usr/lib</code>	Các tệp thư viện của các chương trình người sử dụng
<code>/usr/include</code>	Các tệp include chuẩn của C
<code>/usr/man</code>	Các tài liệu trực tuyến
<code>/usr/log</code>	Các tệp nhật ký người sử dụng và đăng nhập hệ thống.

1.6.7 Các tệp đặc biệt

Có 5 kiểu tệp chính là: file (tệp), directory (thư mục), link (liên kết), named pipe (tuyến dẫn có tên) và device file (tệp thiết bị). Kiểu tệp là kiểu chúng ta đã quen thuộc. Sau đây chúng ta sẽ xét các kiểu tệp khác:

Các liên kết: DOS cung cấp các lệnh như subst và join làm cho một thư mục xuất hiện như một ổ đĩa tách biệt. Linux hỗ trợ liên kết ở mức mạnh hơn, nó cho phép một tệp tồn tại trong một thư mục trong khi bản thân nó thực sự tồn tại trong một thư mục khác. Chúng ta sẽ xét các liên kết kỹ hơn trong phần 2.2.3.

Các tệp thiết bị: Linux quản lý các thiết bị ngoại vi qua các tệp thiết bị, nó mã hoá các thiết bị ngoại vi qua các tệp thiết bị. Khi bạn muốn làm việc với một thiết bị, thay vì làm việc trực tiếp qua các cổng, bạn có thể làm việc với các tệp thiết bị tương ứng với nó.

1.6.8 Shell

Chương trình Shell của Linux cung cấp các chức năng giao tiếp giữa người sử dụng và hệ thống. Nó có nhiều điểm giống với COMMAND.COM trên DOS:

- Các biến môi trường: Là các biến tham số hệ thống. Các biến này có thể thay đổi theo từng người sử dụng hoặc theo từng phiên làm việc của người sử dụng.
- Các chương trình chạy khi khởi động.
- Dấu nhắc hệ thống.

Tuy nhiên shell trong Linux mạnh hơn COMMAND.COM rất nhiều, nó cung cấp rất nhiều khả năng khác mà DOS không có. Chúng ta sẽ trở lại vấn đề này trong phần 2.3.1.

1.6.9 Hệ thống cửa sổ X

Hệ thống cửa sổ X cung cấp giao diện người dùng đồ hoạ cho Linux. Nó gần giống với Microsoft Windows, tuy nhiên hơi khó dùng hơn và cho phép người sử dụng quyền điều khiển và khả năng cấu hình nhiều hơn.

1.6.10 TCP/IP và mạng

Ngay từ đầu, Linux đã là một hệ điều hành mạng, do đó nó tương thích với giao thức TCP/IP và các giao thức trên mạng khác. Hệ thống NFS và NIS cho phép người sử dụng có thể đăng nhập và làm việc từ bất kỳ máy nào trên mạng như là trên một máy duy nhất.

Chương 2. LÀM QUEN VỚI LINUX, CÁC LỆNH CƠ BẢN

Trong chương này, chúng ta sẽ làm quen với Linux và một số khái niệm căn bản của Linux:

- Bắt đầu và kết thúc phiên làm việc.
- Các lệnh cơ bản về hệ thống file trên Linux.
- Các khái niệm cơ bản của Linux

2.1 LÀM QUEN VỚI LINUX

2.1.1 Bắt đầu phiên làm việc

Khi bắt đầu bật máy tính (terminal), trên màn hình sẽ hiện lên thông báo mời:

```
Linux login:
```

Trên Linux, mỗi người sử dụng đều phải có một *account* của riêng mình và chỉ được quyền làm những gì mà *account* đó được phép. Nếu bạn chưa có một *account* thì hoặc là bạn đánh `root` như một người quản trị hệ thống, hoặc bạn phải liên hệ với *root* (người quản trị hệ thống) để có thể tạo ra một *account* cho riêng bạn. Khi đã có một *account*, bạn hãy đánh tên *account* của bạn, khi đó bạn sẽ thấy:

```
Password:
```

Và bạn sẽ phải nhập mật khẩu tương ứng với *account* của bạn và kết thúc bằng Enter. Nếu mọi thứ đều suôn sẻ, có nghĩa là nếu tên và mật khẩu đúng thì bạn có thể bắt đầu phiên làm việc của mình. Khi đó trên màn hình sẽ hiện ra thông báo mời:

```
[huynq@may12 huynq]$
```

đó là dấu nhắc của chương trình *shell*-chương trình điều khiển các dòng lệnh.

2.1.2 Kết thúc phiên làm việc

Để kết thúc phiên làm việc, bạn ấn Ctrl-D hoặc gõ lệnh:

```
$ exit
```

2.1.3 Thay đổi mật khẩu

Mật khẩu đảm bảo an toàn cho mỗi phiên làm việc. Bạn có thể thay đổi mật khẩu bằng lệnh

```
$ password
```

Lệnh này sẽ hỏi bạn nhập vào mật khẩu mới và sau đó yêu cầu bạn gõ lại lần thứ hai để tránh sự nhầm lẫn. Tiêu chuẩn cơ bản là mật khẩu phải dài ít nhất 6 ký tự, chứa ít nhất hai chữ cái và phải khác với tên account ít nhất 3 ký tự.

2.1.4 Các console ảo

Như các hệ điều hành đa nhiệm khác, Linux cho phép bạn có thể làm nhiều việc tại một thời điểm. Phần lớn người dùng Linux khi muốn làm nhiều việc một lúc đều dùng hệ thống X Window. Trước khi chúng ta làm quen với X Window, chúng ta có thể làm việc với các console ảo.

Để làm việc trên các console ảo, bạn giữ phím Alt và ấn một trong 8 phím chức năng từ F1 đến F8. Mỗi khi bạn ấn một phím chức năng, bạn sẽ thấy một màn hình hoàn toàn mới với một dấu mời *login*. Bạn có thể login vào nhiều console ảo khác nhau và có thể chuyển đổi từ console nào sang console kia để làm nhiều việc khác nhau.

Bạn có thể chạy hệ thống X Window và hệ thống này được ngầm định là console 7. Khi làm việc trên X Window, nếu bạn thấy Alt + phím chức năng kích hoạt một menu thay vì việc chuyển đổi giữa các console, bạn hãy dùng Ctrl + Alt + phím chức năng.

2.2 CÁC LỆNH CƠ BẢN

2.2.1 Cách dùng lệnh của Linux

Cú pháp cơ bản để chạy một dòng lệnh trên Linux như sau:

```
$ tên_lệnh [-tùy chọn] [đối số 1] [đối số 2] ...  
[đối số n]
```

Ví dụ về dòng lệnh:

1. Dấu phân cách	
ĐÚNG	SAI
\$ ls -l /usr	\$ ls -l/usr
\$ ls -s /home	\$ ls-s /home
2. Thứ tự	
ĐÚNG	SAI
\$ cat -s file	\$ cat file -s
\$ cat file	\$ file cat
\$ ls -a	\$ -a ls
3. Nhiều tùy chọn	
ĐÚNG	SAI
\$ ls -l -d -t	\$ ls -l-d-t
\$ ls -ldt	\$ ls -l d t
4. Nhiều tham số	
ĐÚNG	SAI
\$ cat file1 file 2	\$ cat file1file2
	\$ cat file1, file2

☞ Chú ý:

- Khi có nhiều tùy chọn có thể viết gộp lại: `ls -l -d -t <=> ls -ldt`.
- Dấu cách, thứ tự, các tùy chọn và các tham số thường tuân theo nguyên tắc trên.

Ví dụ: `wc` là lệnh đếm và hiển thị số dòng, số từ và số kí tự củ một tệp. Ta có thể chạy lệnh này với các tham số khác nhau như sau:

```
$ wc /etc/passwd
```

```
32      37      1139 /etc/passwd
```

```
$ wc -l /etc/passwd
```

```
32      /etc/passwd
```

```
$ wc -lc /etc/passwd /etc/group
```

```
32      1139 /etc/passwd
```

```
15      337 /etc/group
```

```
47      1476 total
```

Các thông báo lỗi khi gọi lệnh:

- Nếu lệnh không tồn tại hoặc không tìm thấy:

```
$ vidu
```

```
vidu: not found
```

- Nếu cú pháp của lệnh bị gõ sai:

```
$ wc -m /etc/group
```

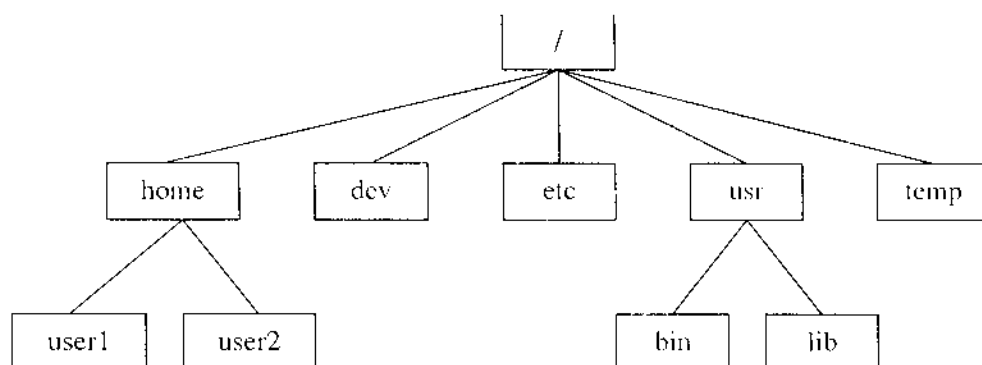
```
usage: wc [-clw] [name ...]
```

2.2.2 Các lệnh về thư mục

Giống như MS-DOS, các file trên Linux cũng được tổ chức theo cấu trúc cây phân cấp thư mục. Trên Linux, bạn sẽ tìm thấy một thư mục có tên là */home* trong đó chứa các file của người dùng. Mỗi một người sử dụng (user) có một thư mục con tương ứng trong thư mục */home* gọi là thư mục cá nhân (home directory). Chẳng hạn nếu tên user là *nghuy* thì thư mục cá nhân tương ứng sẽ là */home/nghuy*. Tất nhiên bạn có thể tạo các thư mục con của riêng bạn trong thư mục cá nhân của bạn. Thư mục cá nhân được ký hiệu ngắn gọn bằng dấu ~, điều này có nghĩa là *~/vanban* là thư mục con *vanban* trong thư mục cá nhân của bạn.

Các thư mục được phân cách bởi dấu gạch chéo /. Thư mục / được gọi là thư mục gốc. Cấu trúc cây thư mục cơ sở của Linux được bố trí như sau:

Chúng ta sẽ xem xét kỹ hơn về thư mục và các file trong phần hệ thống file.



Hình 2.1. Cấu trúc cây cơ sở của Linux

Khi bạn bắt đầu phiên làm việc, hệ thống sẽ đặt bạn vào thư mục cá nhân của bạn (home directory). Để xem thư mục hiện thời đang làm việc, bạn dùng lệnh *pwd* (print working directory):

```
$ pwd
/home/nghuy
```

- **Lệnh chuyển thư mục**

Để chuyển đổi thư mục làm việc, bạn dùng lệnh *cd*, lệnh này gần giống như lệnh *cd* trên DOS:

```
$ cd /usr/bin
$ pwd
/usr/bin
$ cd
$ pwd
/home/nghuy
```

như bạn đã thấy, lệnh *cd* không có tham số sẽ đưa bạn trở về thư mục cá nhân.

- **Lệnh tạo thư mục**

Bạn có thể tạo thư mục bằng lệnh *mkdir* tương tự như trong DOS:


```
$ mkdir vanban
```

hoặc bạn có thể dùng tên đầy đủ:

```
$ mkdir vanban
```

và bạn kiểm tra lại:

```
$ cd vanban
$ pwd
/home/nghuy/vanban
```

Cũng giống như trong DOS, bạn có hai chuỗi ".." và "." đại diện cho thư mục cha và thư mục hiện tại:

```
$ pwd
/home/nghuy/vanban
$ cd ..
$ pwd
/home/nghuy
```

- ***Lệnh xoá thư mục***

Để xoá thư mục rỗng bạn dùng lệnh *rmdir*:

```
$ pwd
/home/nghuy
$ rmdir vanban
```

Lệnh *rmdir* với lựa chọn *-r* (recursive) sẽ xoá toàn bộ thư mục và bên trong nó:

```
$ rmdir vanban -r
```

2.2.3 Các lệnh về tệp

- *Lệnh xem thư mục*

Lệnh *ls* tương tự như lệnh *dir* trong DOS:

```
$ ls /home
```

Bạn có thể dùng lựa chọn *-a* để xem tất cả các file hay các thư mục ẩn:

```
$ cd
$ ls -a
.                  .bashrc
.fvwmrc
..                 .emacs
.xinitrc
.bash-history     .exrc
```

Các file có tên bắt đầu bởi dấu chấm là các file ẩn. Rất nhiều chương trình sử dụng các file ẩn dành cho các lựa chọn của người dùng. Chẳng hạn bạn có thể đặt các lệnh vào trong file *.Xdefaults* để thay đổi cách sử dụng giao diện X Window của các chương trình.

Bạn có thể dùng lựa chọn *-l* để xem chi tiết hơn:

```
$ ls -l
-rw-r--r-- 1 nguyen user1 2451 Feb  7 07:30 strudel
-rw-r--r-- 1 nguyen user1 4025 Feb 10 19:12 borscht
drwxr-xr-- 2 nguyen user1 512 Feb 10 19:12 backups
.
```

Chúng ta sẽ xét kỹ hơn về các thuộc tính của file trên Linux trong phần sau.

- *Xem nội dung một file*

Để xem nội dung của một file, chúng ta có thể dùng một trình soạn thảo chẳng hạn:

```
$ emacs .bashrc
```

hoặc bạn có thể xem nhanh một file bằng lệnh *cat*:

```
$ cat .bashrc
```

Nếu file quá dài khi dùng lệnh *cat* nội dung của file sẽ bị trôi. Khi đó chúng ta có thể dùng Shift + PageUp hoặc Shift + PageDown để xem. Chúng ta cũng có thể xem nội dung một file bằng lệnh *more*, lệnh này sẽ cho phép chúng ta xem từng màn hình một:

```
$ more .bashrc
```

Lệnh *more* cho phép nhiều lựa chọn rất mạnh. Chẳng hạn bạn có thể tìm kiếm một chuỗi trong file bằng cách: ấn /, gõ chuỗi cần tìm và ấn Enter.

Một dạng khác của lệnh *more* là lệnh *less*. Lệnh này còn có nhiều khả năng mạnh hơn cả lệnh *more*, chẳng hạn bạn có thể đánh dấu một vị trí trong file và quay trở lại sau đó.

- **Các liên kết tượng trưng (symbolic links)**

Các liên kết tượng trưng của một file có thể coi là các file giả trở đến một file khác. Nếu bạn làm việc trên một file liên kết, hệ thống sẽ dẫn thông minh để trở đến file thật khi cần thiết. Các liên kết tượng trưng trên Linux làm việc gần giống với các shortcut trên Windows nhưng mạnh hơn rất nhiều.

Ví dụ chúng ta có file *prog.1.1*, chúng ta muốn tạo ra một liên kết tên là *prog* trở đến file này. Ta gõ lệnh sau:

```
$ ln -s prog.1.1 prog
```

Bây giờ bạn đã có file *prog*, nếu bạn chạy nó tức là bạn đã chạy file *prog.1.1*. Bạn hãy xem các thuộc tính của file này:

```
$ ls -l prog
lrwxrwxrwx 2 nquy user1 8 Feb 7 07:30 prog -
> prog.1.1
```

kí tự l đầu dòng có nghĩa là một liên kết (link) và dấu -> chỉ ra file thực mà *prog* trỏ tới.

- **Sao chép file**

Bạn dùng lệnh *cp* để sao chép các file, cú pháp cơ bản của lệnh như sau:

```
$ cp <nguồn> <đích>
```

Ví dụ, để copy file *vidu.txt* vào thư mục *vanban* ở trong thư mục cá nhân của bạn, bạn gõ lệnh:

```
$ cp vidu.txt vanban
```

Bạn có rất nhiều lựa chọn đối với lệnh *cp*, sau đây là một số lựa chọn thường gặp:

-i : interactive (tương tác). Nếu file đích đã có sẵn thì máy sẽ hỏi bạn có copy đè hay không.

-r : recursive (truy hồi). Copy tất cả cây thư mục.

-f : force (bắt buộc). Copy bất kể file đích có hay không. Chủ dùng trong trường hợp bạn đã chắc chắn biết rõ file nguồn và file đích.

-v : verbose (chi tiết). Hiện ra quá trình copy file.

Bạn có thể tìm hiểu rõ hơn lệnh sao chép file bằng lệnh *man*:

```
$ man cp
```

- **Chuyển file**

Linux cung cấp lệnh *mv* để chuyển các file từ vị trí này sang vị trí khác. Ví dụ bạn chuyển file *vidu.txt* từ thư mục cá nhân vào thư mục *vanban* bên trong thư mục này:

```
$ mv vidu.txt vanban
```

```
$ ls vanban
```

```
vidu.txt
```

Lệnh *mv* cũng có các lựa chọn như -i, -f, -v như lệnh *cp* ở trên.

Bạn cũng có thể dùng lệnh *mv* để đổi tên file. Ví dụ bạn muốn đổi tên file *vidu.txt* thành *vidu1.txt*:

```
$ mv vidu.txt vidu1.txt
```

- **Xoá file**

Để xoá file, bạn cũng dùng lệnh *rm* như đối với các thư mục, ví dụ:

```
$ rm vidu.txt
```

Bạn có thể dùng lệnh *rm* với các lựa chọn như: -i, -f, -v. Ví dụ:

```
$ rm -i vidu.txt
rm: remove 'vidu.txt'?
```

Bạn có thể xoá các file bằng cách sử dụng dấu *** tương tự như trong DOS, ví dụ:

```
$ rm vid*
```

Bạn cũng có thể xoá nhiều file một lúc, ví dụ:

```
$ rm vidu1.txt vidu2.txt
```

Bạn có thể dùng lệnh xoá file cùng với các lựa chọn -i, -f, -v, -r giống như lệnh *cp*. Để xoá một thư mục với *rm*, bạn bắt buộc phải sử dụng lựa chọn -r.

Chú ý
Hãy cẩn thận! Lệnh <i>rm</i> rất mạnh, bạn có thể xoá toàn bộ hệ thống của bạn! Nếu bạn là root và bạn chỉ cần gõ lệnh <i>rm -rf</i> thì lệnh này sẽ xoá sạch mọi thứ trong hệ thống của bạn!

Đối với các thư mục, bạn có thể dùng một lệnh an toàn hơn đó là lệnh *rmdir*. Với lệnh này bạn sẽ không có quyền sử dụng lựa chọn truy hồi, do đó một thư mục không rỗng sẽ không được xóa.

2.2.4 Các lệnh khác

- **Lệnh date:**

Lệnh **date** hiển thị ngày, giờ hệ thống, giờ GMT, ...

```
$ date
Fri Jan 12 17:21:14 CDT 1996
$ date -u
Fri Jan 12 23:21:21 GMT 1996
$
```

☞ **Chú ý:**

Người quản trị mạng có thể thay đổi ngày hệ thống bằng lệnh này với các tùy chọn sẵn có.

- **Lệnh cat**

Chức năng: Nối, hiển thị nội dung các tệp

Cú pháp:

```
cat [<các tùy chọn>] <tên tệp>
```

Ví dụ 1:

```
$ cat myfile
Lệnh này hiển thị dữ liệu trong tệp myfile
```

Ví dụ 2:

```
$ cat myfile1 myfile2
Nội dung tệp myfile1
Nội dung tệp myfile2
```

- **Lệnh who**

Xác định những ai đang sử dụng hệ thống:

```
$ who
root      lft0      Jan 17 15:11
minh      pts/2     Jan 17 15:32 (aixclient)
root      pts/1     Jan 17 15:27 (9.24.118.1)
minh      pts/2     Jan 17 16:11 (aixclient)
$
```

Xác định xem bạn đang sử dụng với tên gì:

```
$ who am i
minh      pts/2     Jan 17 16:11 (aixclient)
$
```

- **Lệnh man:**

Hiển thị thông tin trợ giúp trực tuyến theo từng trang

Cú pháp:

```
man <tên lệnh/hàm>
```

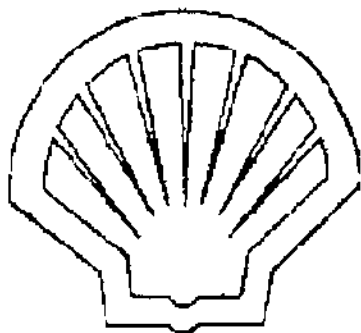
Ví dụ:

```
$ man ls
$ man man
```

- ✓ Hiển thị mô tả, cú pháp và ví dụ về lệnh được chỉ ra.
- ✓ Là một cách để có được tài liệu về lệnh.
- ✓ Trong một vài trường hợp, nếu lệnh được đánh vào với tùy chọn không hợp lệ, cú pháp của lệnh được hiển thị.

2.3 CÁC KHÁI NIỆM CƠ BẢN

2.3.1 Shell



Như chúng ta đã nói ở phần trước, khi bắt đầu phiên làm việc chúng ta sẽ làm việc với shell của Linux. Điều này cũng xảy ra khi chúng ta mở một cửa sổ xterm trong X Window. Shell sẽ dịch và thực hiện mọi lệnh bạn gõ vào từ bàn phím. Trên Linux hiện có vài loại shell như:

- bash : Bourne Again shell. Shell mạnh nhất và thông dụng nhất trên Linux.
- csh : C shell. Hầu như tương thích với bash.
- ksh : Korn shell. ksh shell rất thông dụng trên các hệ thống Unix.
- sh : Bourne shell. Đây là shell nguyên thủy.
- zsh : Z shell. shell mới nhất hiện nay, tương thích với Bourne shell.

Để nhận biết được đang làm việc trên shell nào, chúng ta dùng lệnh:

```
$ echo $SHELL
```

2.3.2 Sự tự động hoàn thành lệnh

Linux hỗ trợ khả năng tự động hoàn thành lệnh (word completion) rất hữu dụng bằng phím Tab. Khả năng này sẽ giúp bạn tiết kiệm được rất nhiều thời gian. Chẳng hạn bạn gõ dòng lệnh sau và không gõ Enter:

```
$ cd /usr/inc
```


sau đó bạn gõ phím Tab. Khi đó shell sẽ tự động thêm "lude" để hoàn thành tên thư mục /usr/include. Bây giờ bạn có thể gõ Enter để thực hiện lệnh một cách bình thường.

Nếu có nhiều khả năng để lựa chọn, hầu hết các shell sẽ bỏ qua. Tuy nhiên bash còn cho phép bạn lựa chọn bằng cách ấn Tab hai lần. Chẳng hạn, bạn gõ:

```
$ cd usr/l
```

và ấn hai lần phím Tab, bash sẽ in ra một bảng lựa chọn các khả năng có thể để bạn chọn:

```
lib    local
```

2.4 TÊN FILE VÀ TÊN THƯ MỤC

Không giống như các hệ điều hành khác như DOS và Windows, Linux cho phép bạn đặt tên các file và các thư mục khá thoải mái. Trừ dấu gạch chéo (/) phân chia thư mục, hầu hết các kí tự đều có thể dùng để đặt tên. Tuy nhiên bạn chú ý khi dùng các kí tự đặc biệt như *, ", - sẽ dễ dẫn đến nhầm lẫn. Các tên file có thể dài đến 255 kí tự. Không giống với hệ điều hành DOS sử dụng dấu chấm (.) để phân chia phần đuôi file, tên file trên Linux có thể chứa nhiều dấu chấm. Tuy nhiên hầu hết những người sử dụng Linux đều tuân theo một số quy ước như trên bảng 2.1. Một vài ngôn ngữ lập trình yêu cầu tên file phải tuân theo các quy ước này.

Bảng 2.1. Một số quy ước về phần đuôi tệp

Phần đuôi	ý nghĩa	Ví dụ	Yêu cầu
.c	Chương trình C	ls.c	Có
.f	Chương trình FORTRAN	math.f	Có
.pl	Chương trình Perl	hose.pl	Không
.h	File include	term.h	Không
.d, .dir	là thư mục	recipes.d, ecipes.dir	Không
.gz	file nén GNV	foo.gz	Có
.Z	file nén	term.h.Z	Có
.zip	file nén PKZIP	book.zip	Có

2.5 CHUYỂN HƯỚNG ĐẦU RA

Linux cung cấp khả năng chuyển hướng đầu ra tương tự trên DOS nhưng mạnh hơn rất nhiều. Nếu bạn đặt dấu > và một tên file đằng sau một lệnh bất kỳ, phần đầu ra của lệnh này sẽ được chuyển đến file xác định bởi tên đã cho. Ví dụ để ghi lại danh sách file của lệnh ls vào file ls_output, bạn làm như sau:

```
$ ls /usr/bin > ~/ls_output
```

Nếu file đã tồn tại, nó sẽ bị xoá. Để tránh điều này, trên bash bạn có thể làm như sau:

```
$ noclobber=1
```

Một cách khác để tránh ghi đè là bạn dùng dấu >> thay vì >. Khi đó nếu file chưa có thì nó sẽ được tạo ra, nếu file đã có thì nó sẽ được ghi nối tiếp vào đuôi.

Hầu hết các chương trình trên Linux đều có hai đầu ra, một đầu được gọi là đầu ra chuẩn, đầu kia gọi là lỗi chuẩn. Shell mặc định đầu ra chuẩn là 1, lỗi chuẩn là 2. Ví dụ bạn có thể chuyển hướng lỗi chuẩn khi dùng trình dịch C gcc để dịch file vd.c như sau:

```
$ gcc vd.c 2> error_msg
```

Nếu bạn không xác định rõ đầu ra chuẩn thì đầu ra chuẩn sẽ được mặc định là màn hình.

Cuối cùng, nếu bạn không muốn đầu ra chuẩn hiện ra màn hình, bạn có thể chuyển hướng nó đến một file đặc biệt là /dev/null. Ví dụ, lệnh sau đây sẽ ghi lại lỗi chuẩn ra file error_msg và bỏ đầu ra chuẩn đi:

```
$ gcc vd.c 2>error_msg >/dev/null
```

2.5.1 Chuyển hướng đầu vào

Không những bạn có thể chuyển hướng đầu ra, bạn có thể chuyển hướng đầu vào chuẩn của các lệnh. Để chuyển hướng đầu vào chuẩn của lệnh, bạn dùng ký hiệu <, cũng tương tự như việc chuyển hướng đầu ra, đầu vào bây giờ cũng là một file có tên file được chỉ ra đằng sau dấu <. Ví dụ, lệnh cat là lệnh để xem nội dung một file, thay vì việc gõ lệnh cat với tham số:

```
$ cat vidu.txt
Cong hoa xa hoi chu nghia Viet Nam.
Doc lap tu do hanh phuc.
```

Bạn có thể dùng chuyển hướng đầu vào cho lệnh *cat*:

```
$ cat < vidu.txt
Cong hoa xa hoi chu nghia Viet Nam.
Doc lap tu do hanh phuc.
```

2.5.2 Các lệnh của Linux

Linux cung cấp tập lệnh với số lượng rất lớn và người dùng có thể tự thêm các lệnh khác. Đó là điểm hoàn toàn khác của Linux với các hệ điều hành khác. Trên Linux, mỗi một lệnh tương ứng với một file. Ví dụ, lệnh *ls* tương ứng với một file nằm trong thư mục */usr/bin*. Do vậy, thay vì gõ *ls*, bạn có thể gõ đầy đủ đường dẫn:

```
$ /usr/bin/ls
```

Điều này làm cho Linux trở nên linh hoạt và rất mạnh. Để cung cấp một tiện ích mới, người quản trị hệ thống chỉ cần cài đặt nó vào trong thư mục chuẩn chứa các lệnh.

Một trong những lỗi thường gặp khi bạn đánh một lệnh là thông báo không tìm thấy lệnh "Not found". Vấn đề có thể là do bạn không đặt lệnh của bạn vào danh sách thư mục mà shell sẽ tìm kiếm. Bạn có thể xem danh sách này bằng lệnh:

```
$ echo $PATH
/usr/local/bin:usr/bin:usr/X11R6/bin:/bin:/usr/lib/
java/bin:\
/usr/games:/usr/bin/TeX:.
```

Lệnh này cho phép chúng ta xem nội dung của *biến môi trường* PATH của Linux. Các thư mục được phân cách nhau bởi dấu *:*. Bạn có thể thêm vào danh sách tìm kiếm, chẳng hạn thư mục */usr/sbin* như sau:

```
$ export PATH=$PATH:/usr/bin
```

2.5.3 Các lệnh được thực hiện nền

Do Linux là một hệ điều hành đa nhiệm, chúng ta có thể thực hiện nền một lệnh (running in background), có nghĩa là khi lệnh thực hiện, chúng ta vẫn còn dấu nhắc của shell và chúng ta vẫn có thể tiếp tục làm các việc khác. Ví dụ:

```
$ gcc vd.c &
[1] 21345
```

Dấu & sẽ làm cho lệnh được thực hiện nền. Số [1] chỉ ra số hiệu được gán cho lệnh, 21345 là số ID của *tiến trình* (có nghĩa là lệnh khi thực hiện). Nếu bạn chạy càng nhiều tiến trình nền, hệ thống sẽ càng chậm.

Nếu bạn muốn chuyển một tiến trình nền về bình thường, bạn dùng lệnh *fg*, tiếp theo là số hiệu lệnh hoặc số ID. Trong trường hợp trên, bạn dùng lệnh:

```
$ fg %1
```

hoặc

```
$ fg 21345
```

Bạn cũng có thể dùng hẳn một tiến trình bằng lệnh *kill*:

```
$ kill %1
```

hoặc

```
$ kill 21345
```

2.5.4 Giúp đỡ bằng lệnh man

Như chúng ta đã biết ở phần trước, Linux cung cấp hệ thống trợ giúp trực tuyến rất mạnh gọi là các trang trợ giúp. Khi bạn cần thông tin về một lệnh trong Linux, bạn dùng lệnh *man*, ví dụ:

```
$ man ls
```

Các trang trợ giúp được chia thành các phần khác nhau: các lệnh của người sử dụng trong phần 1, các lệnh hệ thống trong phần 2, các lệnh quản trị hệ thống nằm trong phần 8, v.v... Bạn có thể gọi từng trang riêng biệt khi cần trợ giúp về một lệnh:

```
$ man 1 ls
```

Để tìm tên một lệnh, bạn có thể dùng lệnh *apropos*, ví dụ:

```
$ apropos edit
```

Lệnh này sẽ in ra tất cả các lệnh có tên chứa chuỗi bạn nhập vào.

Trong X Window, bạn có thể dùng lệnh tương tự *xman*.

Các chương trình của GNU cũng có các trang trợ giúp tương tự với lệnh *info*. Chẳng hạn, để đọc các trang *info* của lệnh *find*, bạn đánh lệnh:

```
$ info find
```

2.5.5 Các thuộc tính của file

Sở hữu file và quyền truy nhập là cơ sở của bảo mật trong các hệ thống đa người dùng. Tính bảo mật là một trong những ưu điểm của Linux so với các hệ điều hành khác.

Ví dụ khi bạn đánh lệnh sau, bạn sẽ nhận được một thông báo từ chối nếu bạn không phải là người quản trị hệ thống:

```
$ cd /root
```

```
bash: /root: Permission denied
```

Một cách để có thể tránh được các thông báo từ chối là bạn dùng lệnh *su* để truy nhập vào hệ thống như là root (người quản trị hệ thống):

```
$ su
```

```
Password: (Gõ password của root)
```

```
# cd /root
# pwd
/root
```

Tất cả các file và thư mục đều được sở hữu bởi người tạo ra nó. Ví dụ nếu bạn tạo ra file vidu.txt trong thư mục người dùng của bạn thì file này sẽ thuộc quyền sở hữu của bạn. Điều đó có nghĩa là, bạn có thể xác định ai có quyền đọc, ghi file hoặc chạy nó nếu nó là một chương trình chạy được.

Có ba loại quyền truy nhập cơ bản đó là: đọc (reading), ghi (writing) và thực hiện (executing). Bởi vì mọi người dùng của hệ thống khi được tạo ra đều được đặt vào trong một nhóm (group) nào đó nên chúng ta cũng có thể xác định xem nhóm nào có thể đọc, ghi hay thực hiện file của chúng ta.

Chúng ta hãy lấy ví dụ file vidu.txt của bạn:

```
$ ls -l vidu.txt
-rw-rw-r-- 1      nh Huy group1 150 Mar 19 08:08
vidu.txt
```

Chúng ta có thể nhìn thấy ai có quyền đọc (r) hoặc ghi (w) file, cũng như ai là người đã tạo ra file (nh Huy) và người này thuộc vào nhóm này (group1).

Các thông tin về quyền truy nhập có nghĩa như sau:

rwX	rwX	rwX			
↓	↓	↓	người sử dụng khác	r	= read (đọc)
↓	↓	↓	nhóm	w	= write (ghi)
			người sở hữu	x	= execute (thực hiện)

Đối với tệp thông thường:

r	==>	có thể xem nội dung của tệp
w	==>	có thể thay đổi hay xóa nội dung của tệp
x	==>	có thể sử dụng tên tệp như tên lệnh

Đối với thư mục:

r	==>	có thể liệt kê (ls, li) các tệp bên trong thư mục
w	==>	có thể tạo/xóa các tệp bên trong thư mục (đòi hỏi cả quyền x)
x	==>	có thể vào trong thư mục (chuyển tới, truy cập các tệp hay thư mục con)

trong trường hợp trên:

```
-   (rw-)   (rw-)   (r--)   1   nh Huy   group1
|       |       |       |
type owner   group   others
```

Thành phần đầu tiên là kiểu của file, kiểu có thể là:

d : thư mục

- : file bình thường

l : liên kết file

Các thành phần tiếp theo là quyền truy nhập, có ba loại quyền truy nhập như sau:

r : file có thể đọc

w : file có thể ghi

x : file có thể thực hiện

Nếu chúng ta thấy một dấu gạch - trong quyền truy nhập của người sở hữu (*owner*), của nhóm (*group*) hay của những người sử dụng khác (*others*), điều đó có nghĩa là không được phép có quyền truy nhập tương ứng.

Đối với file `vidu.txt` ở trên, đây không phải là một chương trình, do đó người sử dụng không có quyền thực hiện nó. Nhóm (*group*) `group1` có quyền đọc và ghi file và những người sử dụng khác (*others*) chỉ có quyền đọc file này mà thôi.

Bạn có thể dùng lệnh `chmod` để thay đổi quyền truy nhập của một file. Ví dụ, bạn muốn cho phép những người sử dụng khác thêm quyền ghi, bạn gõ lệnh như sau:

```
$ chmod o+w vidu.txt
```

Để kiểm tra kết quả bạn đã làm, bạn gõ:

```
$ ls -l vidu.txt
-rw-rw-rw-   1 nh Huy group1  150 Mar 19 08:08
vidu.txt
```

Khi bạn đặt quyền truy nhập `o+w`, có nghĩa là bạn thêm cho những người sử dụng khác (*others*) quyền ghi. Bây giờ bạn muốn cấm quyền truy nhập của tất cả những người khác (trừ bạn), bạn gõ lệnh:

```
$ chmod go-rw vidu.txt
```

và khi đó kết quả sẽ như sau:

```
-rw----- 1 nguy group1 150 Mar 19 08:08
vidu.txt
```

Đặt quyền truy nhập *go-rw* có nghĩa là huỷ bỏ (-) quyền đọc và ghi (r,w) của nhóm và những người sử dụng khác (g và o). Sau đây là các khả năng có thể :

Các thực thể:

- u : chủ
- g : nhóm
- o : những người sử dụng khác
- a : tất cả (u, g, và o)

Quyền truy nhập:

- r : quyền đọc
- w : quyền ghi
- x : quyền truy nhập

Hành động:

- + : thêm quyền truy nhập
- : huỷ bỏ quyền truy nhập
- = : làm cho nó trở thành quyền truy nhập duy nhất.

Chúng ta hãy thử huỷ bỏ tất cả quyền truy nhập:

```
$ chmod a-rw vidu.txt
```

Bây giờ chúng ta hãy xem ta có thể truy nhập được vào file này không:

```
$ cat vidu.txt
cat: 'vidu.txt': Permission denied
```


Chúng ta không có quyền truy nhập vào file vidu.txt nữa! Nhưng vì chúng ta là sở hữu của file, rất may là ta còn có thể được lại được quyền truy nhập:

```
$ chmod u+rw vidu.txt
$ cat vidu.txt
Cong hoa xa hoi chu nghia Viet Nam.
Doc lap tu do hanh phuc.
```

Sau đây là một số ví dụ thông dụng mà bạn có thể sử dụng với *chmod*:

```
g+w: Thêm quyền ghi cho nhóm
o-rwx: Huỷ bỏ mọi quyền truy nhập của những người
sử dụng khác
u+x: Cho phép người sử dụng thực hiện file
a+rw: Cho phép tất cả người sử dụng đọc và ghi
file
ug+r: Cho phép người sử dụng và nhóm quyền đọc
file
g=rx: Nhóm chỉ được phép đọc hoặc thực hiện (không
được ghi).
```

Nếu chúng ta thêm lựa chọn *-R*, chúng ta có thể thay đổi quyền truy nhập cho toàn bộ cây thư mục.

Chúng ta chú ý rằng, đối với các thư mục thì việc "thực hiện" (x) chính là việc vào thư mục để xem.

Ví dụ, để cho phép tất cả mọi người sử dụng đều có thể truy nhập vào thư mục *public* trong thư mục của bạn, bạn gõ lệnh:

```
$ chmod -R a+rw public
```

Nhưng nếu bạn không cho phép những người sử dụng khác quyền thực hiện (x) vào thư mục *public* thì sẽ không có ai có thể vào được thư mục này, trừ khi họ biết rõ tên file ở trong thư mục *public* mà họ cần truy nhập.

Ví dụ nếu bạn bỏ quyền thực hiện của thư mục *public* bằng lệnh:

```
$ chmod a-x public
```

Khi chúng ta muốn vào trong thư mục này sẽ có thông báo lỗi:

```
$ cd public
```

```
bash: public: Permission denied
```

Vừa rồi chúng ta đã biết cách đặt quyền truy nhập bằng chế độ ký hiệu (symbolic mode). Ngoài ra chúng ta cũng có thể dùng chế độ tuyệt đối (absolute mode). Đôi khi bạn sẽ thấy chế độ này tiện lợi hơn khi sử dụng chế độ ký hiệu.

Trong chế độ tuyệt đối, quyền truy nhập của mỗi một kiểu người sử dụng (u, g và o) bao gồm 3 bit đại diện cho các quyền đọc, ghi và thực hiện (r, w, x) tương ứng. Trên hình 2.2 minh họa các bit này.

Người sử dụng (u)			Nhóm (g)			Người sử dụng khác (o)		
r	w	x	r	w	x	r	w	x
400	200	100	40	20	10	4	2	1

Hình 2.2. Các bit quyền truy nhập trong chế độ tuyệt đối

Ví dụ bạn muốn đặt quyền đọc ghi file vidu.txt cho bạn và cấm mọi quyền đối với những người khác, bạn chỉ cần cộng các số tương ứng bên dưới quyền tương ứng trên hình 2.2 : $400+200=600$. Lệnh `chmod` sẽ như sau:

```
$ chmod 600 vidu.txt
```

Bây giờ bạn muốn đặt quyền đọc, ghi và thực hiện cho bạn, quyền đọc và ghi cho nhóm và quyền đọc cho những người sử dụng khác thì bạn cộng tương ứng như sau:

$$400+200+100 + 40+20 + 4 = 764$$

và lệnh `chmod` khi đó sẽ là:

```
$ chmod 764 vidu.txt
```

2.5.6 Các file cấu hình thực hiện

Đặt cấu hình là một trong những khả năng rất mạnh của Linux. Các file cấu hình trên Linux thường có phần đuôi là *rc* (resource configuration) và bắt đầu bằng dấu chấm (.), có nghĩa là nếu bạn dùng lệnh *ls* với các lựa chọn thông thường thì bạn sẽ không nhìn thấy các file này. Các file cấu hình thường nằm trong thư mục cá nhân của bạn. Các file này có thể có hoặc không, nếu không có, các chương trình tương ứng sẽ sử dụng các mặc định. Sau đây là một số các file cấu hình thông dụng:

.bashrc

File cấu hình của shell bash. Đây là một file *shell script*, tức là nó chứa các lệnh và lập trình của shell. File này sẽ được chạy khi bạn bắt đầu phiên làm việc hoặc khi bạn mở một cửa sổ *xterm* trong X Window. Sau đây là ví dụ về một file cấu hình rất đơn giản của shell bash:

```
# .bashrc

PS1='\u$'      # dấu mời có chứa tên user

HISTSIZE=50    # history gồm 50 lệnh

IGNOREEOF=1    # cấm ấn Ctrl-D để kết thúc

stty erase "^H"  # có thể xóa bằng phím
BackSpace

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# cấu hình cho PVM:
# append this file to your .bashrc to set path
according to # machine type.  you may wish to
use this for your own

# programs (edit the last part to point to a
different
```

```
# directory f.e. ~/bin/_$PVM_ARCH.
#

if ! [ -L $HOME/pvm3 ]; then
    ln -s /public/pvm3 $HOME/pvm3
fi

export PVM_ROOT=$HOME/pvm3
export PVM_DPATH=$PVM_ROOT/lib/pvmd
export PVM_ARCH=LINUX
export PATH=$PATH:$PVM_ROOT/lib          #
generic
```

.bash_profile

File cấu hình của shell bash. File này chỉ được chạy khi bạn bắt đầu phiên làm việc. Nếu bạn muốn nó được chạy khi bạn mở một *xterm* trong X Window, bạn mở *xterm* với lệnh *:xterm -ls*.

.emacs

File cấu hình cho trình soạn thảo Emacs. File này chứa các hàm LISP.

.exrc

File cấu hình cho trình soạn thảo vi.

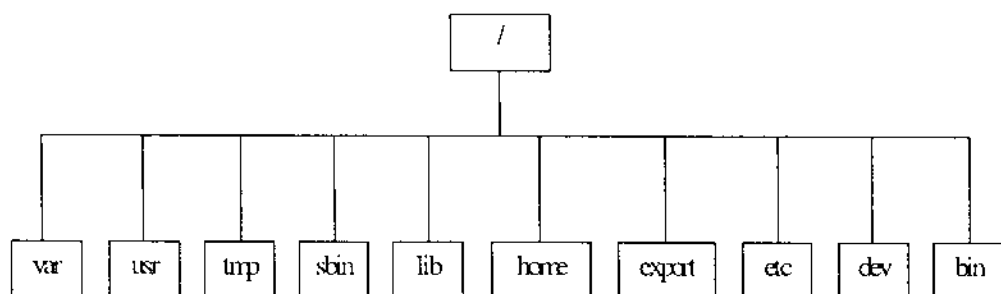
Xdefaults

File cấu hình cho các chương trình sử dụng hệ thống X Window. Mỗi dòng của file xác định một tài nguyên (thường là tên một chương trình và các thuộc tính của nó) cùng với giá trị mà nó sẽ nhận. Chúng ta sẽ xét file này kỹ hơn trong chương 4.

.xinitrc

File cấu hình cho hệ thống X Window. File này chứa các lệnh shell được chạy khi bạn bắt đầu một phiên làm việc X Window.

2.5.7 Các thư mục quan trọng



Chúng ta đã biết thư mục `/home`, nơi chứa các file của người sử dụng. Nếu bạn là người quản trị hệ thống hoặc người lập trình, bạn cần biết một vài thư mục quan trọng. Sau đây là một số thư mục như vậy:

/bin

Chứa các lệnh Linux cơ bản nhất.

/usr/bin

Chứa các lệnh khác.

/usr/sbin

Chứa các lệnh quản trị hệ thống.

/boot

Được sử dụng để lưu các file khi khởi động.

/etc

Chứa các file được sử dụng bởi các hệ thống con như mạng, NFS và thư điện tử. Thông thường chúng chứa các bảng phục vụ mạng, các thông tin đĩa, v.v...

/var

Chứa các file quản trị hệ thống như các file *log* (nhật ký), được rất nhiều chương trình sử dụng.

/var/spool

Nơi lưu trữ tại các file đang được in, các file đang gửi của UUCP, v.v...

/usr/lib

Các thư viện chuẩn như *libc.a*. Khi bạn liên kết (*link*) một chương trình, chương trình *link* luôn luôn tìm kiếm các thư viện ở đây (các thư viện được xác định bởi lựa chọn *-l*)

/usr/lib/X11

Chứa các thư viện của người sử dụng X Window, các font chữ, các file tài nguyên mẫu và các phần quan trọng khác của X Window. Thư mục này thường được liên kết tới thư mục */usr/X11R6/lib/X11*.

/usr/include

Chứa các file include của C, chẳng hạn *stdio.h*

/usr/src

Chứa mã nguồn của các chương trình trong hệ thống.

/usr/local

Các chương trình và dữ liệu đã được người quản trị hệ thống cài đặt cục bộ.

/etc/skel

Các file cấu hình mẫu mà bạn có thể đưa vào thư mục cá nhân của những người sử dụng mới.

2.5.8 Các tiến trình

Khái niệm chiếm vị trí trung tâm trong hệ điều hành Linux là khái niệm tiến trình (*process*). Một tiến trình trong Linux là một chương trình độc lập đang chạy cùng với tập các tài nguyên riêng của nó. Ví dụ, chúng ta có thể hướng đầu ra của một chương trình vào một file trong khi shell vẫn đưa các kết quả của chương trình này ra màn hình. Lý do mà shell và chương trình đó có thể cùng gửi kết quả ra các vị trí khác nhau đó là chúng là hai tiến trình độc lập với nhau.

Trong Linux, các tài nguyên trên máy tính như bộ nhớ, đĩa và bộ vi xử lý đều được điều khiển bởi một chương trình duy nhất đó là nhân hệ điều hành (kernel). Tất cả các chương trình khác đều là các tiến trình.

Trước khi bạn bắt đầu phiên làm việc, cửa sổ làm việc (*terminal*) của bạn được điều khiển bởi một tiến trình tên là *getty*. Sau khi bạn đã bắt đầu phiên làm việc, tiến trình *getty* này mất đi và *terminal* của bạn bây giờ được điều khiển bởi shell của bạn. Shell sẽ tạo ra một tiến trình mới mỗi khi bạn gõ vào một lệnh.

Để xem các tiến trình đang chạy của bạn, bạn có thể dùng lệnh *ps*. Bạn sẽ thấy danh sách các tiến trình bạn đang chạy, trong đó một trong những tiến trình chính là lệnh *ps* bạn vừa gõ.

```
[huyngq@may12 huyngq]$ ps
  PID TTY          TIME CMD
   951 pts/0        00:00:00 bash
   953 pts/0        00:00:00 emacs
   955 pts/0        00:00:00 ps
[huyngq@may12 huyngq]$
```

Trong đó:

PID : Số ID của tiến trình (Process ID)

TTY : Terminal điều khiển

TIME : Thời gian sử dụng CPU

CMD : lệnh (command).

Trường đầu tiên PID là số ID duy nhất của tiến trình. Bạn có thể dùng số ID này để dừng một tiến trình bằng lệnh *kill*:

```
$ kill số_ID.
```

Chẳng hạn khi bạn muốn dừng tiến trình *emacs* ở trên , bạn gõ lệnh:

```
$ kill 953
```

Bạn cũng có thể xem tất cả các tiến trình đang chạy trên hệ thống bằng lựa chọn a (*all*) hoặc x (cho các tiến trình không gắn với tiến trình nào):

```
$ ps aux
```

Nếu có quá nhiều tiến trình, để tránh màn hình bị trôi bạn có thể dùng lệnh *more*:

```
$ ps aux | more
```

hoặc có thể ấn Shift-PgUp, Shift-PgDown để xem lại (chức năng này chỉ có từ RedHat 6.1 trở lên). Sau đây là ví dụ về lệnh *ps*:

```
$ ps aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.1	0.1	1104	460	?	S	15:26	0:03	init {3}
root	2	0.0	0.0	0	0	?	SW	15:26	0:00	{kflushd}
root	3	0.0	0.0	0	0	?	SW	15:26	0:00	{kupdate}
root	4	0.0	0.0	0	0	?	SW	15:26	0:00	{kpiod}
root	5	0.0	0.0	0	0	?	SW	15:26	0:00	{kswpd}
root	6	0.0	0.0	0	0	?	SW<	15:27	0:00	{mdrecoveryd}
bin	277	0.0	0.1	1204	480	?	S	15:27	0:00	portmap
root	293	0.0	0.1	1088	460	?	S	15:27	0:00	/usr/sbin/apmd-p
root	310	0.0	0.2	1256	524	?	S	15:27	0:00	ypbind (master)
root	315	0.0	0.2	1296	644	?	S	15:27	0:00	ypbind (slave)
root	378	0.0	0.0	0	0	?	SW	15:27	0:00	{rpciod}
root	397	0.0	0.2	1152	556	?	S	15:27	0:00	syslogd -m 0
root	408	0.0	0.2	1424	756	?	S	15:27	0:00	klogd
daemon	424	0.0	0.1	1128	484	?	S	15:27	0:00	/usr/sbin/atd
root	440	0.0	0.2	1304	600	?	S	15:27	0:00	cron
root	456	0.0	0.1	1124	484	?	S	15:27	0:00	inetd
root	477	0.0	0.1	1092	380	?	S	15:27	0:00	rpc.rquotad
root	488	0.0	0.2	1364	708	?	S	15:27	0:00	rpc.mountd --no-n
root	499	0.0	0.0	0	0	?	SW	15:27	0:00	{nfsd}
root	500	0.0	0.0	0	0	?	SW	15:27	0:00	{nfsd}
root	501	0.0	0.0	0	0	?	SW	15:27	0:00	{nfsd}
root	502	0.0	0.0	0	0	?	SW	15:27	0:00	{nfsd}
root	503	0.0	0.0	0	0	?	SW	15:27	0:00	{nfsd}
root	504	0.0	0.0	0	0	?	SW	15:27	0:00	{nfsd}

root	505	0.0	0.0	0	0 ?	SW	15:27	0:00	[nfsd]
root	506	0.0	0.0	0	0 ?	SW	15:27	0:00	[nfsd]
root	528	0.0	0.1	1132	440 ?	S	15:27	0:00	gpm -t ps/2
xfs	545	0.0	0.5	2184	1304 ?	S	15:27	0:00	xfs -droppriv -da
root	561	0.0	0.3	2400	912 ?	S	15:27	0:00	smbd -D
root	572	0.0	0.3	1984	968 ?	S	15:27	0:00	nmdb -D
root	608	0.0	0.1	1076	384 tty2	S	15:27	0:00	/sbin/mingetty tt
root	609	0.0	0.1	1076	384 tty3	S	15:27	0:00	/sbin/mingetty tt
root	610	0.0	0.1	1076	384 tty4	S	15:27	0:00	/sbin/mingetty tt
root	611	0.0	0.1	1076	384 tty5	S	15:27	0:00	/sbin/mingetty tt
root	612	0.0	0.1	1076	384 tty6	S	15:27	0:00	/sbin/mingetty tt
root	633	0.0	0.4	2228	1164 tty1	S	15:38	0:00	login -- huynq
huynq	634	0.0	0.3	1728	1016 tty1	S	15:39	0:00	-bash
huynq	653	0.0	0.3	1684	836 tty1	S	15:40	0:00	sh /usr/X11R6/bin
huynq	660	0.0	0.2	2296	756 tty1	S	15:40	0:00	xinit /etc/X11/xi
root	661	1.8	2.2	12376	5924 ?	S	15:40	0:49	/etc/X11/X :0 -au
huynq	665	0.0	1.0	5480	2664 tty1	S	15:40	0:00	/usr/bin/gnome-se
huynq	675	0.0	0.7	5560	2012 tty1	S	15:40	0:00	gnome-smproxy --s
huynq	681	0.1	1.0	4200	2612 tty1	S	15:40	0:02	enlightenment -cl
huynq	696	0.0	0.5	2704	1332 ?	S	15:40	0:00	gnome-name-servic
huynq	698	0.0	0.6	2996	1608 tty1	S	15:40	0:00	xscreensaver -no-
huynq	700	0.0	1.9	7648	5148 tty1	S	15:40	0:01	panel --sm-client
huynq	702	0.0	2.1	7696	5416 tty1	S	15:40	0:00	gmc --sm-client-i
huynq	716	0.1	1.3	5988	3392 ?	S	15:40	0:02	gnomepager_applet
huynq	718	0.0	1.2	5932	3220 ?	S	15:40	0:00	gen_util_applet -
root	873	0.0	0.9	3688	2484 ?	S	15:47	0:00	smbd -D
huynq	875	0.2	3.9	11848	10264 tty1	S	15:49	0:05	gimp
huynq	876	0.0	0.8	5212	2284 tty1	S	15:49	0:00	/usr/lib/gimp/1.0
lequan	947	0.0	0.3	1556	824 ?	S	16:08	0:00	/home/lequan/pvm3
huynq	949	0.0	0.9	3652	2536 tty1	S	16:09	0:00	nxterm
huynq	951	0.0	0.3	1728	996 pts/0	S	16:09	0:00	bash
huynq	953	0.0	1.9	6860	4916 pts/0	S	16:09	0:00	emacs
huynq	966	0.0	0.3	2704	1000 pts/0	R	16:23	0:00	ps aux

2.5.9 Các tuyến dẫn (pipes)

Linux đưa ra khái niệm tuyến dẫn (pipes), khái niệm này làm cho Linux trở nên rất mạnh so với các hệ điều hành khác như DOS và Windows. Tuyến dẫn là khái niệm mô tả các kênh dữ liệu vào hoặc ra của một lệnh trên Linux. Các tuyến dẫn được dùng để nối đầu ra của lệnh này với đầu vào của lệnh kia.

Chúng ta hãy lấy lại ví dụ về lệnh ps ở trên:

```
$ ps aux
```

để có thể xem được dễ hơn, chúng ta hãy kết hợp lệnh này với lệnh more hoặc less bằng cách sử dụng khái niệm tuyến dẫn:

```
$ ps aux | less
```

Bạn để ý dấu | là dấu dùng để nối hai lệnh với nhau qua tuyến dẫn. Bây giờ chúng ta có thể xem danh sách của các tiến trình mỗi lần một màn hình. Bạn có thể dùng các phím lên, xuống để xem từng dòng một hoặc dùng các phím B và Space để xem từng màn hình một. Để thoát lệnh less, bạn bấm phím Q.

2.5.10 Nối các lệnh thành chuỗi

Linux cho phép bạn vào nhiều lệnh tại cùng một thời điểm. Để làm điều đó, bạn viết các lệnh liên tiếp nhau, các lệnh được phân cách nhau bởi dấu chấm phẩy. Giả sử chúng ta có hai file văn bản vidu1.txt và vidu2.txt như sau:

```
$ cat vidu1.txt
Cong hoa xa hoi chu nghia Viet Nam.
Doc lap tu do hanh phuc.
$ cat vidu2.txt
Truong Dai hoc Bach khoa Ha noi.
```

Bây giờ chúng ta muốn xem nội dung hai file cùng một lúc, chúng ta hãy sử dụng chuỗi lệnh:

```
$ cat vidu1.txt; cat vidu2.txt
Cong hoa xa hoi chu nghia Viet Nam.
```

Doc lap tu do hanh phuc.

Truong Dai hoc Bach khoa Ha noi.

Chúng ta hãy lấy một ví dụ khác. Linux cung cấp lệnh *date* để xem ngày giờ hệ thống. Giả sử bạn muốn biết bạn sử dụng trình soạn thảo *emacs* trong bao nhiêu thời gian, bạn gõ lệnh như sau:

```
$ date; emacs; date
```

Trên xterm của bạn sẽ hiện ra thời gian bạn bắt đầu soạn thảo:

```
[huynq@may12 huynq]$ date; emacs; date
Sat Apr  8 17:40:22 ICT 2000
```

Khi bạn soạn thảo xong, lệnh *date* thứ hai sẽ cho bạn biết bạn đã soạn thảo trong thời gian bao lâu:

```
[huynq@may12 huynq]$ date; emacs; date
Sat Apr  8 17:40:22 ICT 2000
Sat Apr  8 17:43:15 ICT 2000
[huynq@may12 huynq]$
```

Trong ví dụ trên, chúng ta đã dùng *emacs* trong 3 phút.

Chương 3. HỆ THỐNG X WINDOW

3.1 KHÁI NIỆM X WINDOWS

Hệ thống X Window (chúng ta sẽ gọi là X cho gọn) dựa trên mô hình client-server (khách-chủ). Trong đó, mỗi 'X client' là một chương trình ứng dụng, nó gửi các truy nhập tới các phần cứng đồ họa tới 'X server' như: mở một cửa sổ, hiển thị các dòng văn bản, các ảnh,... 'X server' thông tin tới các client các thao tác của người sử dụng như: di chuyển chuột, nhập dữ liệu từ bàn phím, nhấn các nút điều khiển,...

X còn là một hệ thống đồ họa hướng mạng (network-oriented graphics system), tức là, từ một 'X server', bạn có thể nối (connect) tới một hệ thống X khác qua mạng, rồi chạy các ứng dụng trên đó và hướng việc hiển thị kết quả trên 'X server' của bạn.

Các 'X client' chạy trong X có thể được hiển thị trong một hoặc nhiều cửa sổ (window) trên màn hình, và việc quản lý chúng (như: vẽ lại cửa sổ, thay đổi kích thước,...) không thuộc về 'X server' mà thuộc về chương trình điều khiển cửa sổ riêng (window manager) (là một 'X client' chạy song song với các 'X client' khác).

Hệ thống X có tính an toàn cao, tính mô-đun hoá các chức năng, và thích ứng với nhiều dạng kiến trúc khác nhau.

X Windows có hai kiến trúc cơ bản là: GNOME và KDE. Trong đó, ngoài những chức năng dành cho người sử dụng được cung cấp bởi KDE, GNOME còn cung cấp thêm một số chức năng dành cho việc quản trị hệ thống.

3.2 XINIT

Lệnh *startx* cho phép khởi động hệ thống X Windows. Chương trình *xinit* (qua *startx*) sẽ thực hiện các lệnh trong file 'xinitrc' (là một *script shell* nằm trong thư mục cá nhân) nhằm kích hoạt các 'X client' như: *xterm*, *xclock*, *xload*,... . Và file 'xinitrc' trong trường hợp không có sẽ được thay thế bởi: `/usr/lib/X11/xinit/xinitrc`.

Sau đây là một ví dụ của file xinit:

```
#!/bin/sh
#! Sample .xinitrc shell script

#!Start xterms
xterm -geometry 80*40-10+100 -fg black -bg white &
xterm -geometry -20+10 -fn 7*13bold -fg darkslategray -
bg white &
xterm -geometry -20-30 -fn 7*13bold -fg black -bg white
&

# other useful X clients
oclock -geometry 70*70+5+5 &
xload -geometry 85*60+85+5 &
xbiff -geometry +200+5 &
xsetroot -solid darkslateblue &

# start the window manager
exec fvwm2
```

và sau đây là một số chú thích:

- *-geometry* cho phép định vị kích thước và vị trí của 'X client'.
- *-fg* để chỉ ra màu nền (foreground) của text .
- *-bg* để chỉ ra màu nền (background) của cửa sổ 'xterm'.

Ví dụ: 'xterm -geometry 80*40-10+100 -fg black -bg white' chỉ ra cửa sổ có kích thước: 40 dòng và 80 kí tự trên mỗi dòng; có góc trên bên trái được định vị ở tọa độ (10,100) của màn hình; có chữ màu đen; nền màu trắng

- *-fn* chỉ ra font chữ được sử dụng trong cửa sổ 'xterm'.
- Ví dụ: '-fn 7*13bold' chỉ ra cỡ chữ 7*13, và in đậm.

3.3 GNOME

Ta sẽ tìm hiểu trong phần này các nội dung sau:

- Giới thiệu chung
- Trình quản lí cửa sổ và GNOME
- Trình quản lí cửa sổ 'Enlightenment'
- Desktop
- Folders (các thư mục)
- Trình quản lí File của GNOME
- Thanh Panel
- Tìm trợ giúp (Finding Help)

3.3.1 Giới thiệu chung

GNOME là môi trường làm việc ngầm định ('default desktop environment'), mềm dẻo và thân thiện với người sử dụng của Red Hat Linux 6.1. GNOME có đầy đủ các điểm mạnh của hệ thống Red Hat Linux và sự tiện lợi của GUI. Và để biết chi tiết hơn về GNOME, bạn có thể vào các trang WEB sau:

<http://www.gnome.org>.

Khi bắt đầu một phiên làm việc trong X Window, bạn sẽ nhìn thấy:

- *Desktop* - chính là không gian làm việc chính ('main workspace'). Các cửa sổ, ứng dụng, 'folders', tài liệu ... đều được sắp xếp trên đó, và bạn có thể nhấn đúp chuột lên nó để mở một 'folder' hay chạy một ứng dụng.
- *Folders* - là nơi chứa các liên kết ('links') tới các tài liệu như: các tài liệu đi kèm, các tài liệu trợ giúp, các tham chiếu mà chúng có thể được cài đặt trực tiếp trên máy hoặc trên trang WEB. Ta có thể tạo thêm hoặc xóa các folder khỏi Desktop.
- *Panel* - chính là thanh ngang nằm ở phía dưới của Desktop, chứa biểu tượng của một số ứng dụng. Trong đó, bạn chỉ cần kích chuột vào nút menu chính ('Main Menu') để hiển thị tên tất cả các ứng dụng có trong hệ thống.

GNOME cung cấp cho người sử dụng khả năng thao tác mềm dẻo (trong môi trường WINDOWS) thông qua việc sử dụng 'three-button mouse' (chuột 3 nút) và chức năng 'drag and drop' ('kéo rê' và 'nhả').

3.3.1.1 Dùng chuột:

Nếu chuột chỉ có 2 nút ('two-button mouse') nhưng bạn đã chọn chế độ 'three-button emulation', thì 'middle button' (nút giữa) sẽ được coi bằng việc nhấn đồng thời nút chuột trái và phải.

Việc nhấn nút giữa chuột ở điểm trên desktop sẽ hiện ra menu các ứng dụng trong GNOME (GNOME applications menu), các menu trợ giúp, các menu cấu hình để bạn có thể thay đổi lại môi trường GNOME được ngầm định khi cài đặt.

Nút chuột trái được nhấn khi muốn chọn để mở một mục (item) hoặc kéo nó tới một vị trí khác. Có thể chọn nhiều mục trên desktop bằng cách: nhấn giữ chuột trái cùng với việc di chuyển con trỏ tới các mục đó.

Nút chuột phải được nhấn khi muốn hiển thị menu, trong đó chứa các chức năng mà bạn có thể thực hiện như: tạo thêm các 'desktop items',... Nếu nhấn chuột phải vào vị trí một mục (như một folder hay một ứng dụng nào đó), sẽ hiển thị menu chứa các thao tác mà bạn có thể thực hiện đối với mục ấy, ví dụ như xóa, xem các thuộc tính,...

3.3.1.2 'Drag and Drop'

Bạn có thể 'drag and drop' các item trong GNOME. Tức bạn có thể di chuyển các dữ liệu, các ứng dụng từ vị trí này tới vị trí khác; có thể 'nhả' nó vào trong một folders hoặc trong một ứng dụng; có thể di chuyển, copy chúng.

3.3.2 Trình quản lý cửa sổ và GNOME

Trình quản lý cửa sổ (window manager) là một phần mềm có đảm nhận việc quản lý các cửa sổ trong môi trường X WINDOWS như: di chuyển, vẽ lại,... các cửa sổ nào đó.

Khác với một số các hệ điều hành khác, GNOME không phụ thuộc vào bất kỳ một trình quản lý cửa sổ cụ thể nào, điều đó có nghĩa rằng môi trường desktop sẽ không thay đổi ngay cả khi bạn thay đổi trình quản lý cửa sổ. GNOME và trình quản lý cửa sổ làm việc với nhau theo cách có thể đem lại cho người sử dụng một môi trường làm việc dễ chịu nhất. GNOME sẽ không đảm nhận về vị trí của các cửa sổ, mà lấy những thông tin về vị trí của các cửa sổ thông qua trình quản lý cửa sổ.

Sau đây là các địa chỉ về một số trình quản lý cửa sổ trong GNOME:

- Enlightenment - <http://www.enlightenment.org>
- Icewm - <http://www.kiss.uni-lj.si/~k4fr0235/icewm/>
- Window Maker - <http://www.windowmaker.org>
- FVWM - <http://www.fvwm.org/>
- AfterStep - <http://www.afterstep.org/>

Và bạn có thể tìm trong '[GNOME Software Map](#)' để biết thêm về một số trình quản lý cửa sổ đang được phát triển cho môi trường GNOME.

3.3.3 Trình quản lý cửa sổ 'Enlightenment'

Enlightenment là tên của trình quản lý cửa sổ ngầm định đối với Red Hat Linux. Enlightenment còn cung cấp cho bạn các công cụ để giúp bạn cấu hình trình quản lý cửa sổ theo các yêu cầu của bạn.

3.3.3.1 Công cụ cấu hình Enlightenment

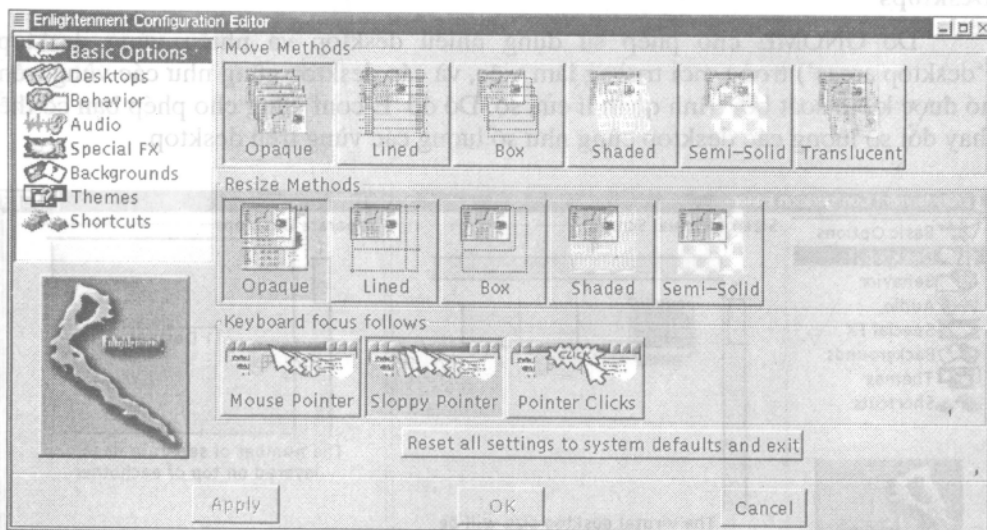
Bạn có thể thực hiện việc cấu hình Enlightenment ('Enlightenment Configuration tool': E-conf) theo một trong hai cách sau:

- Mở 'GNOME Control Center' và trong nhóm Desktop, hãy chọn 'Window Manager Capplet'. Nếu hiện tại, trình quản lý cửa sổ trên máy là Enlightenment, thì bạn có thể chạy E-conf bằng cách nhấn nút 'Run Configuration Tool' của Enlightenment.
- Hoặc, hãy kích nút chuột giữa ở một vị trí trên desktop và chọn 'Enlightenment Configuration' trong menu để chạy E-conf.

Danh sách các kiểm soát của E-conf được hiện ở góc trên bên trái của E-conf, bao gồm các nội dung sau: Basic Options, Desktops, Behavior, Audio, Special FX, Backgrounds, Themes, và Shortcuts.

Basic Options

Lựa chọn này cho phép bạn cấu hình cách thức mà Enlightenment kiểm soát sự di chuyển và cách trình bày, hiển thị, kích hoạt các cửa sổ trên desktop. Sau khi lựa chọn xong, hãy nhấn nút 'Apply'.



Hình 3.1. Các lựa chọn của Enlightenment

Trong mục này, có 3 nội dung chính sau: Move Methods, Resize Methods, và Keyboard.

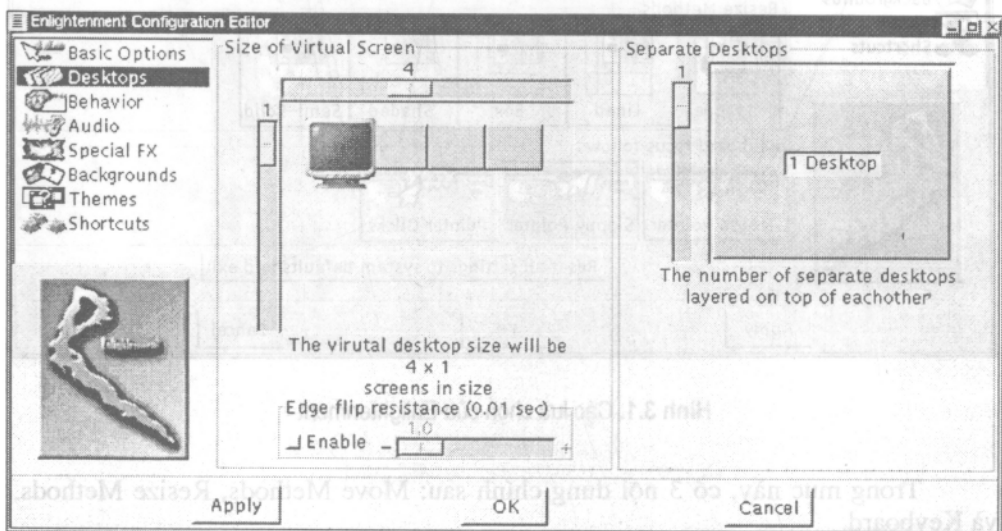
- Move Methods - kiểm soát cách cửa sổ được hiển thị khi ta dùng chuột để di chuyển nó trên màn hình. Và bạn có thể chọn: Opaque, Lined, Box, Shaded, Semi-Solid, hay Translucent.
- Resize Methods - kiểm soát cách các cửa sổ sẽ được hiển thị trong lúc bạn thay đổi kích thước của chúng. Và bạn có thể chọn: Opaque, Lined, Box, Shaded, hay Semi-Solid. Again, depending on your machine speed some of these methods may work slower than others.
- Keyboard focus follows - kiểm soát việc đặt quan tâm ('focus') tới các cửa sổ như việc làm sáng các cửa sổ được kích hoạt (tức được 'focus') và mờ các cửa sổ còn lại. Bạn có thể chọn các đặc tính sau: Mouse Pointer, Sloppy Focus, và Pointer Clicks.

'Mouse Pointer' chỉ làm sáng cửa sổ mà con trỏ chuột đang trỏ trên nó. 'Sloppy Pointer' chỉ làm sáng cửa sổ trong lúc mà con trỏ chuột còn trỏ trên cửa sổ ấy. 'Pointer Clicks' thì chỉ cho phép làm sáng cửa sổ bằng cách phải kích chuột lên cửa sổ ấy.

Bạn có thể đặt Enlightenment trở lại cấu hình ngầm định bằng cách nhấn nút 'Reset all settings to system defaults' và nút 'exit'.

Desktops

Do GNOME cho phép sử dụng nhiều desktop và nhiều vùng desktop ('desktop areas') trong môi trường làm việc, và các desktop cũng như các vùng trên nó được kiểm soát bởi trình quản lý cửa sổ. Đó đó E-conf cũng cho phép bạn có thể thay đổi số lượng các desktop cũng như số lượng các vùng trên desktop.



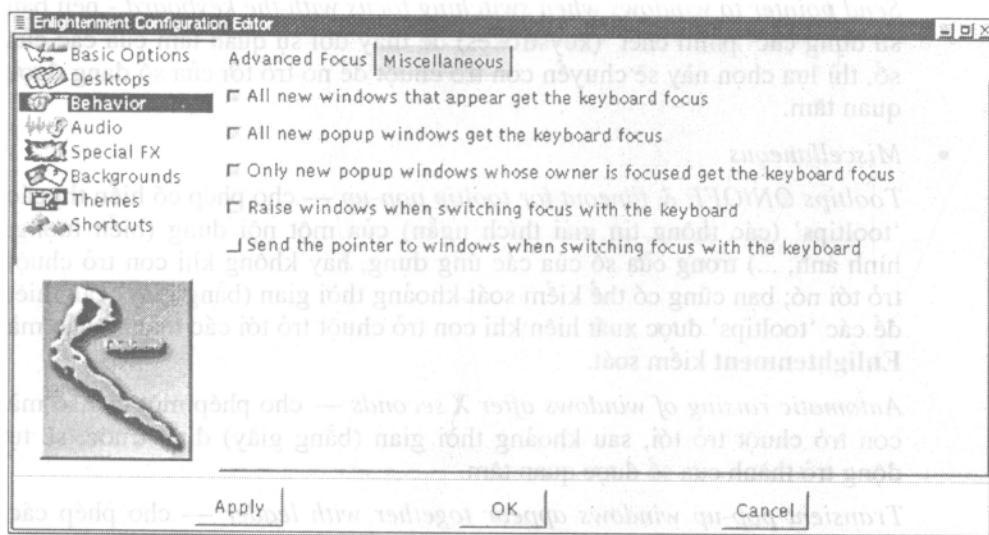
Hình 3.2. Enlightenment Desktops

Trong mục 'Desktops' của E-conf, 'Size of Virtual Screen' là công cụ kiểm soát các vùng trên desktop và cho phép bạn thay đổi số lượng các vùng ấy. Còn 'Separate Desktops' sẽ cho bạn đặt số lượng desktops mà bạn muốn sử dụng trong phiên làm việc của bạn.

Để biết về tình trạng hiện tại của các desktops và các vùng của nó (areas/desktops), hãy chạy 'GNOME Pager'. 'GNOME Pager' sẽ cho bạn biết số lượng các vùng, các desktops mà bạn đang có, cũng như các ứng dụng nào của bạn trong các vùng ấy. 'GNOME Pager' còn cho phép bạn di chuyển từ vùng này tới vùng khác, và từ desktop này tới desktop khác.

Behavior

Mục 'Behavior' của E-conf cho phép bạn kiểm soát các hành vi tương đối đa dạng trong các tình huống của Enlightenment.



Hình 3.3. Enlightenment Behavior

- **Advanced Focus** — cho phép bạn thay đổi các đặc tính liên quan tới việc đặt quan tâm tới các cửa sổ.

All new windows that appear get the keyboard focus — lựa chọn này cho phép bạn có thể làm việc ngay với cửa sổ của ứng dụng (mà bạn vừa cho chạy nó) ngay khi cửa sổ ấy xuất hiện mà không phải dùng chuột để kích hoạt cửa sổ ấy.

All new pop-up windows get the keyboard focus — lựa chọn này chỉ đặt quan tâm tới các cửa sổ pop-up mới. Nếu thuộc tính này được chọn nhưng “*All new windows that appear get the keyboard focus*” không được chọn, thì các cửa sổ pop-up, không phụ thuộc vào vị trí nó xuất hiện, sẽ là các cửa sổ đang được quan tâm (‘focus’), nhưng các cửa sổ của ứng dụng mới được chạy (mà sinh ra nó) thì không. Các cửa sổ pop-up có thể là các thông báo lỗi trong một ứng dụng.

Only new pop-up windows whose owner is focused get the keyboard focus - lựa chọn này chỉ đặt quan tâm tới các cửa sổ pop-up mới nếu như ứng dụng sinh ra nó, trước đó là cửa sổ đang được quan tâm.

Raise windows when switching focus with the keyboard -- nếu bạn sử dụng các ‘phím chết’ (keystrokes) để thay đổi sự quan tâm của các cửa sổ, thì lựa chọn này sẽ chuyển các cửa sổ vừa được quan tâm trở thành các cửa sổ nền (‘foreground’).

Send pointer to windows when switching focus with the keyboard - nếu bạn sử dụng các ‘phím chết’ (keystrokes) để thay đổi sự quan tâm của các cửa sổ, thì lựa chọn này sẽ chuyển con trỏ chuột để nó trỏ tới cửa sổ đang được quan tâm.

- *Miscellaneous*

Tooltips ON/OFF & timeout for tooltip pop-up — cho phép có hiển thị các ‘tooltips’ (các thông tin giải thích ngắn) của một nội dung (biểu tượng, hình ảnh, ...) trong cửa sổ của các ứng dụng, hay không khi con trỏ chuột trỏ tới nó; bạn cũng có thể kiểm soát khoảng thời gian (bằng giây) cần thiết để các ‘tooltips’ được xuất hiện khi con trỏ chuột trỏ tới các thành phần mà **Enlightenment** kiểm soát.

Automatic raising of windows after X seconds — cho phép một cửa sổ mà con trỏ chuột trỏ tới, sau khoảng thời gian (bằng giây) định trước, sẽ tự động trở thành cửa sổ được quan tâm.

Transient pop-up windows appear together with leader — cho phép các cửa sổ pop-up sẽ luôn xuất hiện cùng với cửa sổ sinh ra nó (điều này cho bạn dễ dàng kiểm soát công việc, đặc biệt khi bạn phải làm việc với nhiều cửa sổ cùng lúc).

Switch to where pop-up window appears — cho phép chuyển con nháy chuột (‘mouse cursor’) và quyền kiểm soát bàn phím (‘keyboard control’) tới bất kì cửa sổ pop-up nào khi nó xuất hiện trên màn hình.

Display icons when windows are iconified — lựa chọn này cho phép hiển thị biểu tượng của ứng dụng trên cửa sổ của ứng dụng ấy khi cửa sổ này bị thu nhỏ (minimized).

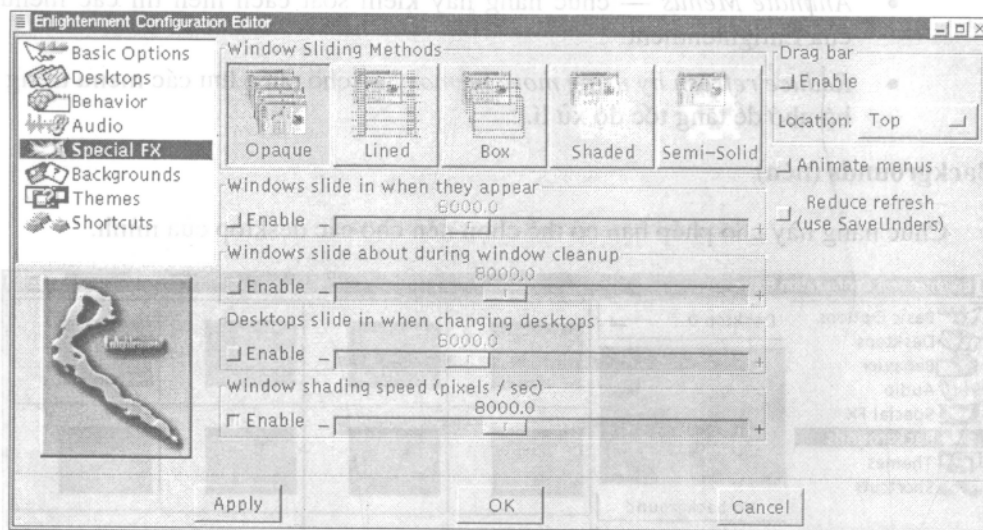
Place windows manually — cho phép bạn tự định vị trí của các cửa sổ được sinh ra.

Audio

Cho bạn đặt các sự kiện liên quan tới âm thanh ('sound events') trong Enlightenment.

Special FX

'Special FX' trong E-conf sẽ kiểm soát các hiệu ứng mà Enlightenment có thể dùng để kiểm soát các cửa sổ.



Hình 3.4. Các hiệu ứng đặc biệt của Enlightenment

- *Windows Sliding Methods* — kiểm soát vùng không gian hiển thị của cửa sổ. Bạn có thể chọn các đặc điểm sau: Opaque, Lined, Box, Shaded, và Semi-Solid.
- *Windows slide in when they appear* — kiểm soát tốc độ trượt (tính bằng số điểm trên một giây) của cửa sổ khi nó được mở lại hay thu nhỏ.

Windows slide about when cleanup in progress — kiểm soát hiệu ứng của cửa sổ trong lúc trượt khi bạn chọn menu 'Cleanup Windows' trong các menu của Enlightenment.

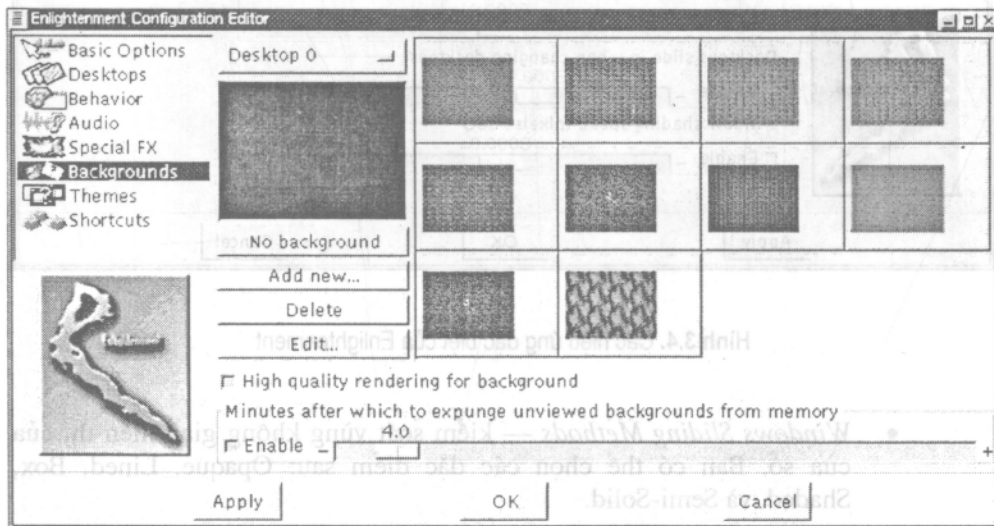
Desktops slide in when changed — trong trường hợp sử dụng nhiều desktops, chức năng này sẽ làm cho desktops trượt rơi ('slide in') khi bạn di chuyển từ desktops tới một desktops khác. Tốc độ 'trượt rơi' của các desktop tính bằng số điểm trong một giây.

Window shading speed — lựa chọn này cho phép bạn tạo bóng của một cửa sổ bằng cách nhấn đúp chuột lên thanh tiêu đề ('title bar') của cửa sổ ấy. Và nhấn đúp chuột lần nữa sẽ trả cửa sổ ấy trở lại trạng thái ban đầu.

- *Drag bar* — cho phép bạn 'slide and peak' vào trong vùng tiếp theo trên Desktop hoặc vào trong Desktop tiếp theo (chỉ trong một số ứng dụng của Enlightenment).
- *Animate Menus* — chức năng này kiểm soát cách hiển thị các menu của Enlightenment.
- *Reduce refresh by using more memory* — cho phép lưu các menu trong bộ nhớ để tăng tốc độ xử lý.

Backgrounds (nền)

Chức năng này cho phép bạn có thể chọn nền cho các desktop của mình.



Hình 3.5. Các nền của Enlightenment

• Đặt nền

Trước tiên bạn phải chọn desktop cần đặt nền. Nút phía trên được đặt là 'Desktop 0' sẽ chỉ tới desktop đang được đặt nền, và nếu như muốn đặt nó cho một desktop khác, bạn có thể nhấn nút ấy và chọn desktop thích hợp.

Tiếp theo, hãy chọn nền cho desktop được chọn. Bạn có thể chọn ảnh hoặc mẫu trong số các mẫu được đưa ra, hoặc chọn nút 'No background'.

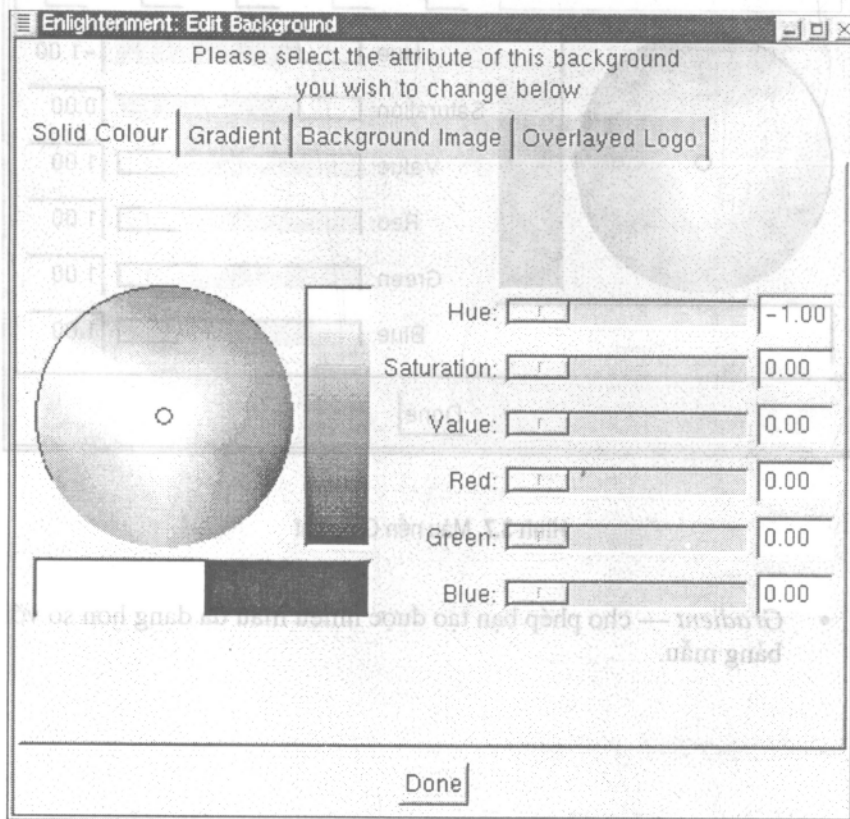
Nếu không thích các mẫu có sẵn đã được đưa ra, thì bạn có thể tạo mẫu nền riêng cho mình.

- Tạo mẫu nền mới

Để tạo một mẫu nền mới để thêm vào trong tập các mẫu đã có sẵn, trước tiên bạn hãy ấn nút 'Add new'. Như vậy, một hộp mới ('new box') sẽ được thêm vào tập mẫu. Ban đầu, hộp này sẽ là hộp trắng.

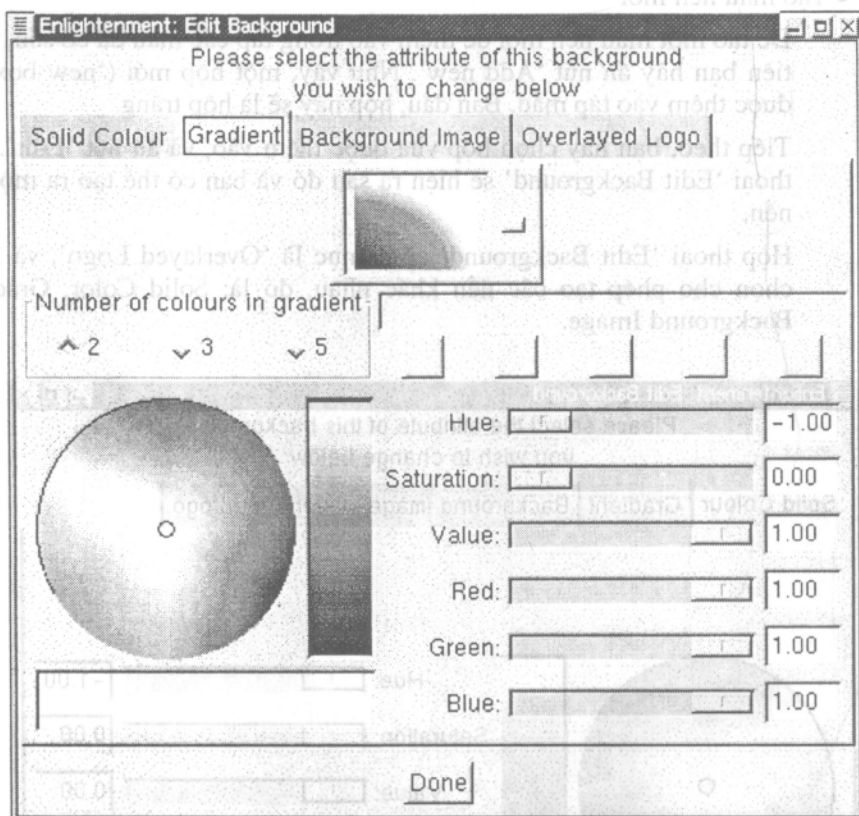
Tiếp theo, bạn hãy chọn hộp vừa được thêm vào, và ấn nút 'Edit'. Hộp thoại 'Edit Background' sẽ hiện ra sau đó và bạn có thể tạo ra một ảnh nền.

Hộp thoại 'Edit Background' có 4 mục là 'Overlaid Logo', và 3 lựa chọn cho phép tạo các nền khác nhau, đó là: Solid Color, Gradient, Background Image.



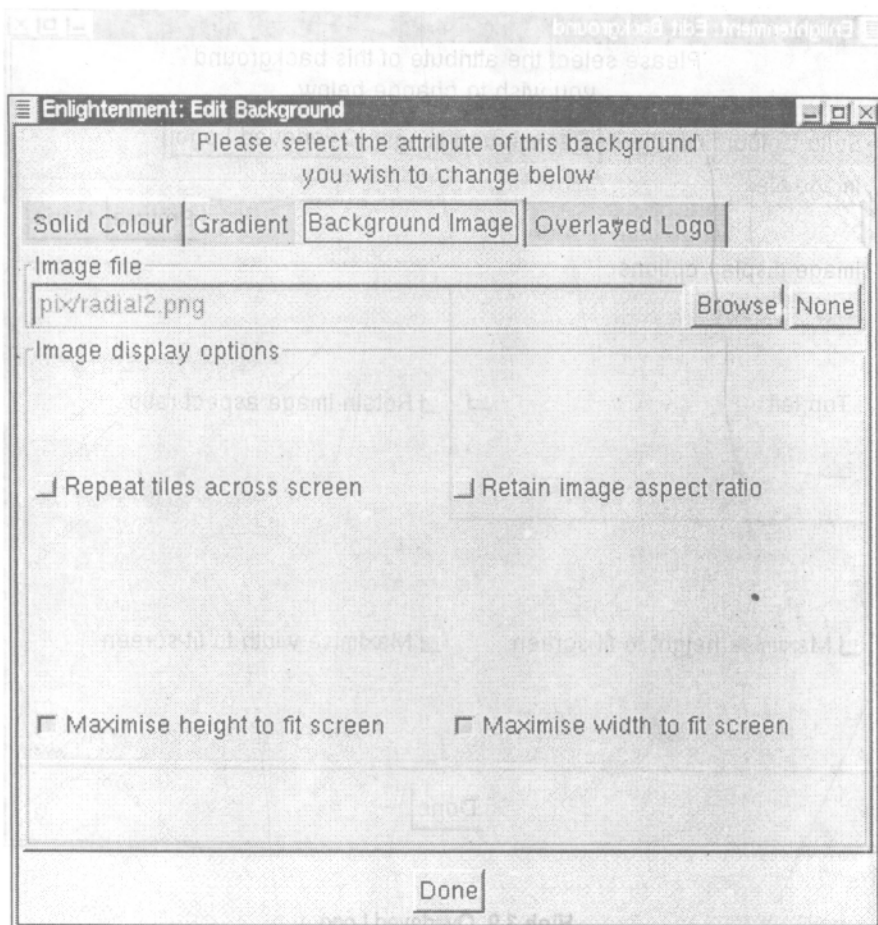
Hình 3.6. Màu nền Solid

- **Solid Color** — cho phép chọn màu để làm màu nền.



Hình 3.7. Màu nền Gradient

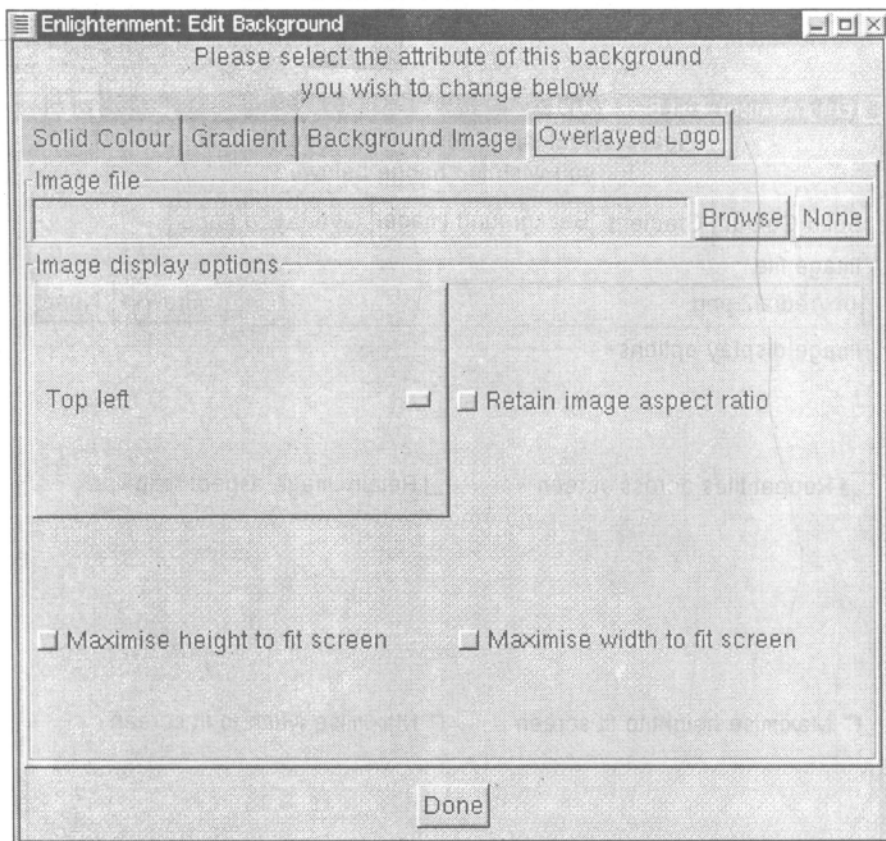
- **Gradient** — cho phép bạn tạo được nhiều mẫu đa dạng hơn so với trong bảng mẫu.



Hình 3.8. Nền ảnh

- *Background Image* — cho phép bạn chọn một ảnh nào đó có trong hệ thống của bạn để làm ảnh nền. Có thể ấn nút 'Browse' để tìm ảnh và nút 'None' để xóa đi một ảnh nào đó được chọn.

Với ảnh đã được chọn, trong trường hợp quá nhỏ so với màn hình, bạn có thể giữ nguyên tỉ số của kích thước ảnh, phóng to chiều dài hay chiều rộng của ảnh cho hợp với màn hình.



Hình 3.9. Overlaid Logo

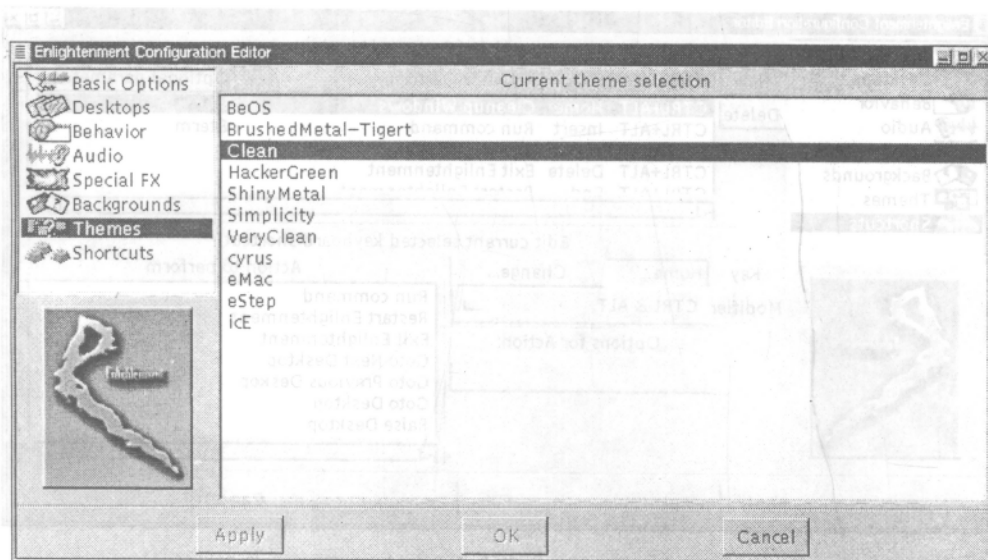
- *Overlaid Logo* — lựa chọn này cho phép bạn bao bọc hình ảnh cho nền mà bạn đã chọn. Và bạn có thể ấn nút 'Browse' để chọn ảnh thích hợp, cũng như ấn nút 'None' để xóa bỏ lựa chọn đó.

Sau khi đã tìm được ảnh cho nền, bạn có thể chọn vị trí để hiển thị nó, cùng vị trí của nó trong bảng mẫu. Bạn cũng có thể giữ nguyên kích thước ảnh, hoặc phóng to ảnh cho khít với chiều dài hay chiều rộng của màn hình.

Sau khi đã hoàn tất việc tạo ảnh nền, hãy ấn nút Done để quay trở lại E-conf.

Themes

Các 'themes' (chủ đề) này được tạo ra bởi 'Raster' và rất nhiều người sử dụng Enlightenment trên thế giới. Một trang Web về các 'themes' của Enlightenment đã được tạo ra là: <http://e.themes.org>.



Hình 3.10. Enlightenment Themes

Một số 'themes' được tạo ra cho các phiên bản cũ của Enlightenment và không thể chạy trên một số phiên bản gần đây của Enlightenment.

Các 'theme' mà bạn tìm được cho Enlightenment, cần được lưu trong thư mục:

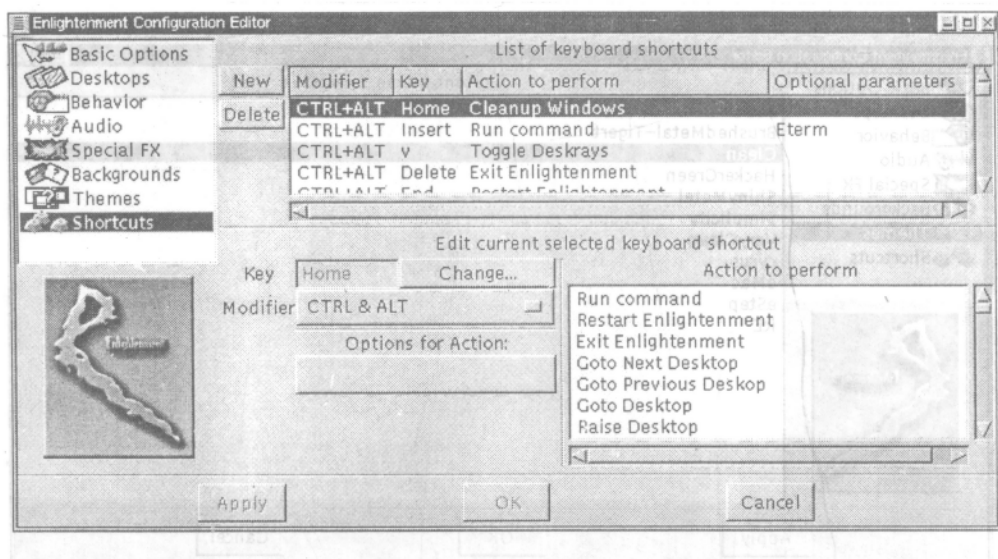
```
~/.enlightenment/themes/
```

Tất cả các file *.etheme* trong thư mục này đều có thể được chọn trong mục 'Themes' của E-conf.

Để thay đổi các 'themes', hãy chọn 'theme' mà bạn muốn trong danh sách các 'themes' dùng được, và ấn nút 'Apply' ở phía dưới của E-conf.

Keyboard Shortcuts

Mục 'Keyboard Shortcuts' cho phép bạn thay đổi các 'shortcuts' đã được dùng để kiểm soát các cửa sổ, thực hiện các lệnh, di chuyển giữa các cửa sổ, giữa các vùng trên desktop và giữa các desktop.



Hình 3.11. Enlightenment Keyboard Shortcuts

‘Keyboard Shortcuts’ được chia thành 2 phần chính là: ‘List of keyboard shortcuts’ và ‘Edit current selected keyboard shortcut’.

Trong ‘List of keyboard shortcuts’, bạn có thể thấy các ‘shortcut’ nào đã được định nghĩa rồi, và chọn chúng để ‘edit’ (tức định nghĩa lại), để xóa nhờ nút ‘Delete’, hoặc tạo thêm các ‘shortcut’ mới nhờ nút ‘New’.

‘Edit current selected keyboard shortcut’ là vùng mà hiện tại bạn đang định nghĩa ‘keyboard shortcut’. Để dùng nó, trước tiên, bạn phải chọn ‘shortcut’ cần được định nghĩa lại, hoặc chọn một ‘shortcut’ mới vừa được thêm vào sau khi chọn nút ‘New’. Sau đó bạn có thể thay đổi các phím (‘keys’) hoặc thậm chí cả chức năng mà ‘shortcut’ này gắn với.

Để thay đổi các phím được sử dụng cho ‘shortcut’, hãy ấn nút ‘Modifier selection’ và chọn một hay nhiều các ‘modifiers’ (các thay đổi) mà bạn muốn, đó là các phím: ALT, CTRL, SHIFT.

Bạn cũng có thể thay đổi phím đang được dùng, bằng cách ấn nút ‘Change’. Sau đó một cửa sổ thông báo sẽ yêu cầu bạn gõ vào phím mà bạn muốn. Hãy gõ nó, và định nghĩa cũ sẽ được thay thế.

Để gắn một ‘shortcut’ với một hành động (action) nào đó, hãy chọn một trong các ‘action’ được liệt kê trong ‘Action’.

Ngoài ra, bạn cũng có thể định nghĩa các cách gõ phím (keystrokes).

3.3.3.2 Enlightenment Menus

Giống như một số trình quản lý cửa sổ, Enlightenment có thể làm việc cùng hoặc không với môi trường desktop, nó cung cấp cho người sử dụng một số chức năng như: chạy ứng dụng, di chuyển giữa các desktop, ...

Enlightenment Desktop Menus

Menu của desktop của Enlightenment sẽ hiện ra khi bạn ấn nút chuột giữa (hoặc ấn cả hai nút chuột trái và phải đồng thời) tại một vị trí nào đó trên desktop.

Chú ý: Các menu này có thể khác nhau nếu như người viết ra các 'theme' cho Enlightenment thay đổi chúng. Nên nếu thấy một số nội dung không khớp với các trình bày sau đây, thì bạn nên đối sánh lại với tài liệu được kèm theo cho 'theme' được sử dụng.

- *GNOME Apps* — mục này giống với các ứng dụng trong 'Main Menu' của GNOME.
- *User Apps* — mục này giống với các ứng dụng mà bạn thêm vào trong menu 'User' trong 'Main Menu' của GNOME.
- *Other Programs* — ở đây là các ứng dụng được đưa vào bởi người viết ra Enlightenment hoặc bởi người viết ra 'theme'. Bạn có thể hoặc không cần tải ('load') các ứng dụng này lên máy của bạn.
- *Desktop* — các chức năng trong menu này cho phép bạn tạo các thay đổi đối với các desktop của bạn.
 - *Cleanup Desktop* — cho phép sắp xếp lại các nội dung trên desktop theo một trật tự được xác định trước.
 - *Go to Next Desktop* — cho phép bạn chuyển tới vùng tiếp theo trong desktop hoặc tới desktop tiếp theo, tùy thuộc vào tình huống lúc đó.
 - *Go to Previous Desktop* — tùy tình huống mà cho phép bạn chuyển tới vùng riêng hoặc tới desktop riêng.
 - *FXRipples*.
- *Themes* — lựa chọn này sẽ cho bạn chọn một theme nào đó đã được cài đặt trên máy của bạn.
- *Enlightenment Configuration* — cho phép chạy E-conf.
- *About Enlightenment* — mục này sẽ hiển thị cửa sổ chứa các thông tin giới thiệu về Enlightenment.
- *Help* — hiển thị phần trợ giúp của Enlightenment.

- *Restart Enlightenment* — cho phép khởi động lại Enlightenment. Nó sẽ không cần được chọn nếu GNOME làm việc như lớp trên của Enlightenment, tuy nhiên, bạn cần sử dụng ‘GNOME Logout’ để thay thế.

Enlightenment Window Menus

Đối với mỗi cửa sổ của Enlightenment, bạn có thể làm hiện ra menu của nó (‘Window Menus’) bằng hai cách sau:

1. sử dụng nút menu của nó (nếu có);
2. hoặc bằng cách ấn phím ‘Alt’ đồng thời với nhấn nút chuột phải lên biên hoặc bên trong cửa sổ ấy.

Sau đây là một số nội dung trong menu của cửa sổ:

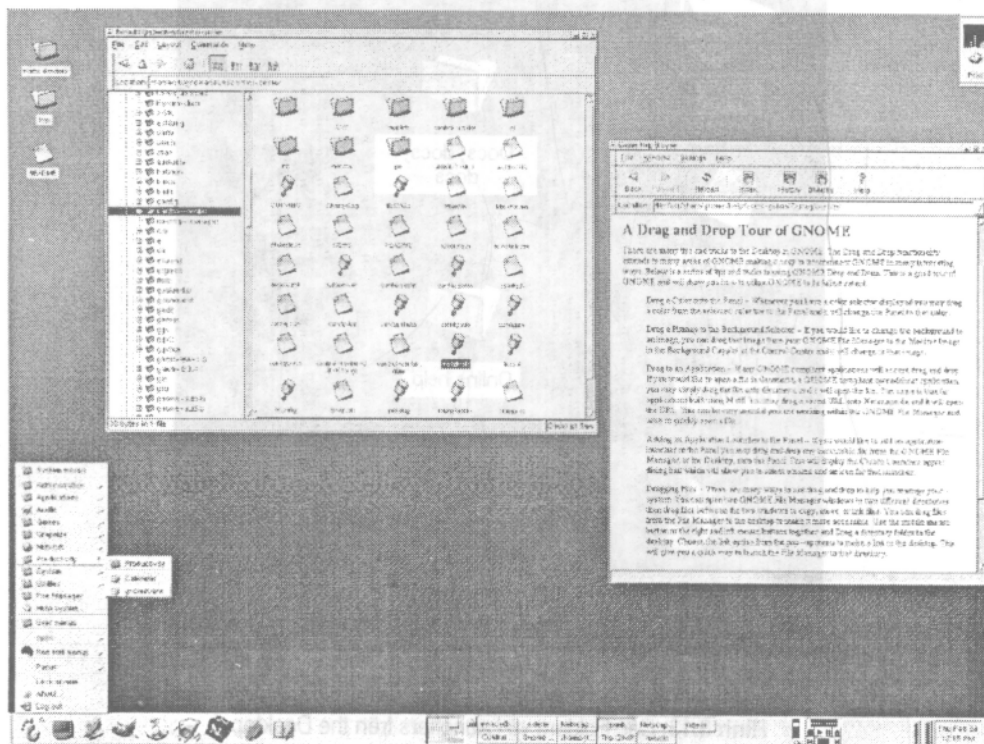
Chú ý: Các menu này có thể khác nhau nếu như người viết ra các ‘theme’ cho Enlightenment thay đổi chúng. Nên nếu thấy một số nội dung không khớp với các trình bày sau đây, thì bạn nên đối sánh lại với tài liệu được kèm theo cho ‘theme’ được sử dụng.

- *Close* — cho phép đóng cửa sổ.
- *Annihilate* — cho phép dừng ứng dụng trong trường hợp bạn gặp lỗi trong quá trình đóng cửa sổ ứng dụng.
- *Iconify* - đưa cửa sổ này trở về dạng biểu tượng.
- *Raise* — làm hiện cửa sổ lên phía trước của tất cả các cửa sổ khác hiện có trên màn hình.
- *Lower* — đưa cửa sổ xuống thấp hơn tất cả các cửa sổ khác hiện có trên màn hình.
- *Shade/Unshade* — làm hiện bóng của cửa sổ này hoặc làm mất bóng của nó.
- *Stick/Unstick* — cho phép cửa sổ nhìn thấy được trên tất cả các desktops và các vùng trên desktop.
- *Desktop* — cho phép di chuyển cửa sổ này tới một vùng khác của desktop. Bạn có thể chọn: Move to Area Right (di chuyển sang phải); Move to Area Left (di chuyển sang trái); Move to Area Above (di chuyển lên phía trên); Move to Area Below (di chuyển xuống phía dưới).
- *Window Size* — cho phép thay đổi kích thước của cửa sổ.

3.3.4 Desktop

3.3.4.1 Giới thiệu chung

Hình 3.12 là một ví dụ của màn hình GNOME:

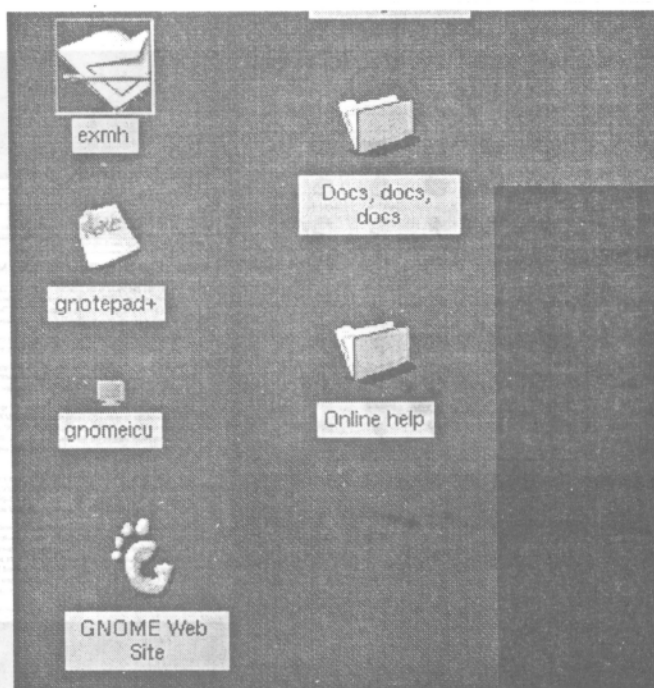


Hình 3.12. Màn hình GNOME

Desktop của GNOME cung cấp mọi chức năng của một desktop của bất kỳ một hệ điều hành truyền thống nào.

- Bạn có thể ‘kéo rê’ các file, các chương trình ứng dụng, các folders để đặt chúng trên desktop; di chuyển vị trí của chúng trên desktop; hoặc xóa chúng khỏi desktop, ...
- Bạn cũng có thể nhấn đúp chuột lên biểu tượng của một item trên desktop để: mở folder nếu item này là một folder, chạy ứng dụng nếu item này là một ứng dụng chạy được, mở một trang WEB nếu biểu tượng này ‘link’ tới trang WEB ấy,....

Ví dụ:



Hình 3.13. Các Folders và Launchers trên the Desktop

Nếu bạn đăng nhập hệ thống với tư cách người quản trị ('root'), bạn có thể gặp một thông báo lỗi nếu như bạn đã không tạo tài khoản người sử dụng trong quá trình cài đặt Red Hat Linux.

3.3.4.2 Một số menu của Desktop

Khi nhấn chuột phải lên vị trí trống trên desktop, sẽ hiện thị một menu dọc chứa một số chức năng sau:

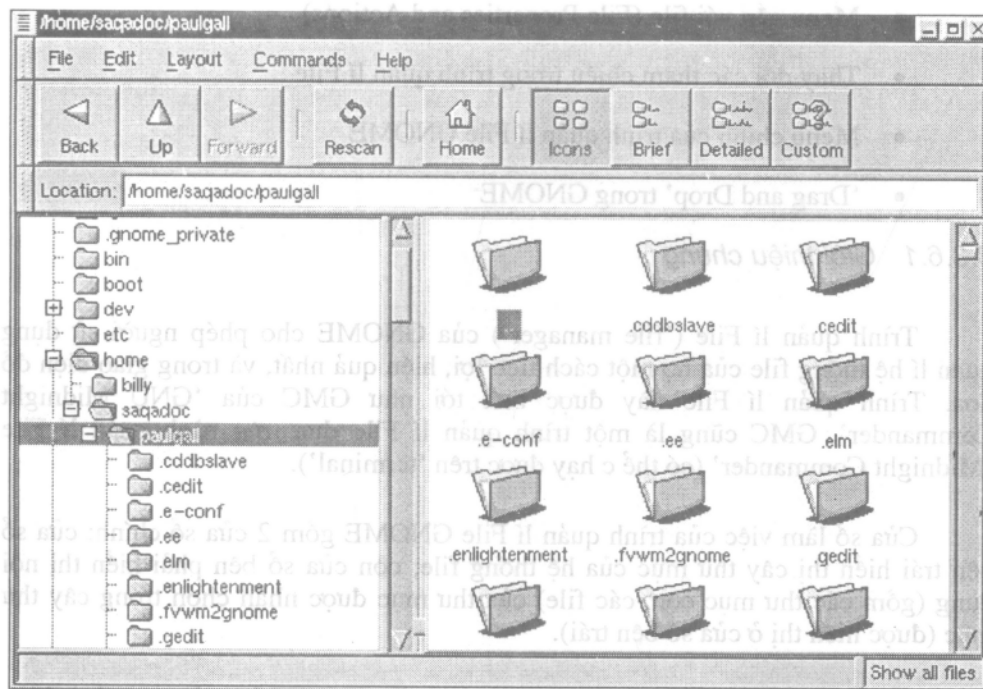
- *New / Terminal* — nhằm tạo thêm một cửa sổ 'terminal' mới trong GNOME.
- *New / Directory* — tạo thêm một thư mục mới trên desktop.
- *New / Launcher* — cho phép tạo thêm một ứng dụng mới ('new application launcher') trên desktop. Khi được chọn, hộp thoại 'Application Launcher' sẽ xuất hiện và cho phép bạn định rõ ứng dụng nào cũng như các thuộc tính của nó.
- *New / [application]* — cho phép khởi động ('start up') một ứng dụng một cách dễ dàng bằng cách kích vào biểu tượng của chúng trong menu 'New'.

- *Arrange Icons* — cho phép sắp xếp các biểu tượng trên desktop một cách tự động.
- *Create New Window* — nhằm đưa ra một cửa sổ ‘GNOME File Manager’ mới trong đó hiển thị nội dung thư mục ‘Home’ của người sử dụng.
- *Recreate Desktop Shortcuts* — cho phép rà soát lại tất cả các thiết bị đã được cài đặt (ngầm định hay không) trên máy, và hiển thị biểu tượng mới cho các thiết bị mới có thể được tìm thấy.
- *Rescan Desktop* — cho phép rà soát lại tất cả các file trong thư mục ‘~/gnome-desktop’ của người sử dụng. Bất kì một file mới nào được tìm thấy cũng sẽ được lưu lại trên desktop ở dạng các biểu tượng.

3.3.5 Các Folder

Các folder trên desktop cho phép lưu trong nó các ứng dụng, các file dữ liệu.

Khi nhấn đúp chuột trên một folder, folder đó sẽ được mở và hiển thị mọi nội dung có trong nó (hình vẽ 3.14).



Hình 3.14. Nội dung của cửa sổ Open Folder

Mặt khác, các folder còn được hiển thị trong 'GNOME File Manager', điều đó cho phép bạn làm việc trong GNOME với tư cách một người sử dụng bình thường hoặc với tư cách của người quản trị hệ thống.

3.3.6 Trình quản lí File của GNOME

Các nội dung được giới thiệu trong phần này như sau:

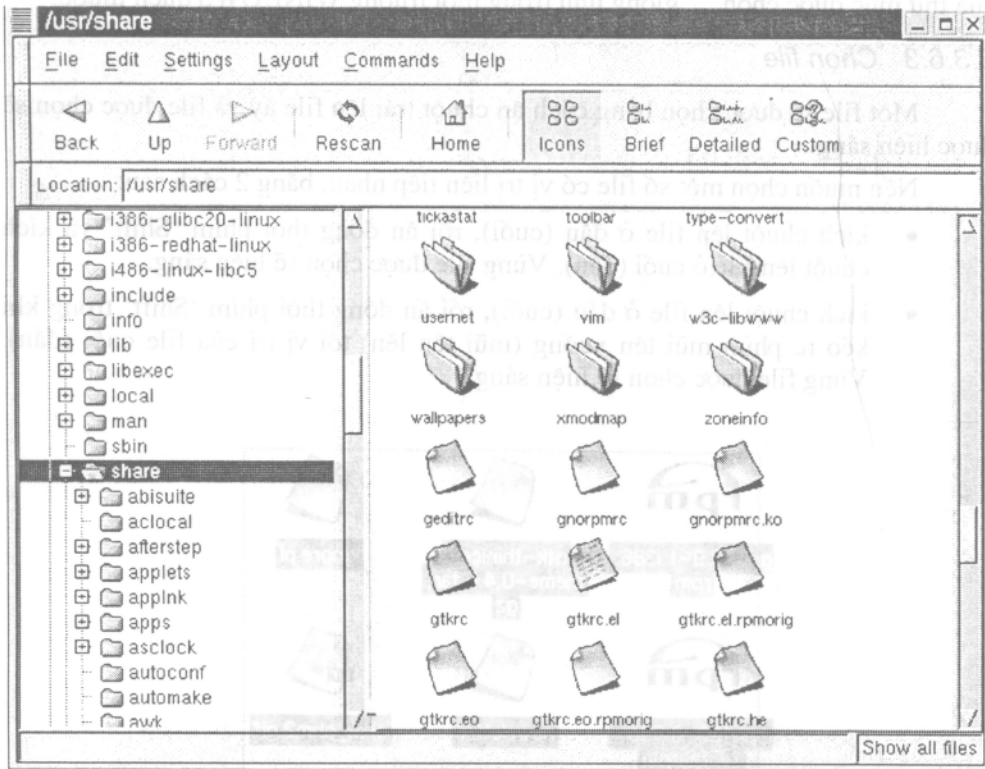
- Giới thiệu chung
- Tìm file
- Chọn file
- Sao chép và di chuyển các file
- Đổi tên file
- Chạy các ứng dụng từ trình quản lí File GNOME
- Menu gắn với file (File Properties and Actions)
- Thay đổi các tham chiếu trong trình quản lí File
- Menu chính của trình quản lí File GNOME
- 'Drag and Drop' trong GNOME

3.3.6.1 Giới thiệu chung

Trình quản lí File ('file manager') của GNOME cho phép người sử dụng quản lí hệ thống file của họ một cách tiện lợi, hiệu quả nhất, và trong giao diện đồ họa. Trình quản lí File này được biết tới như GMC của 'GNU Midnight Commander'. GMC cũng là một trình quản lí File dựa trên trình quản lí File 'Midnight Commander' (có thể chạy được trên 'terminal').

Cửa sổ làm việc của trình quản lí File GNOME gồm 2 cửa sổ chính: cửa sổ bên trái hiển thị cây thư mục của hệ thống file; còn cửa sổ bên phải hiển thị nội dung (gồm các thư mục con, các file) của thư mục được nhấn chọn trong cây thư mục (được hiển thị ở cửa sổ bên trái).

Bạn có thể chọn file, mở các thư mục, ... bằng cách giống như bạn thực hiện trong môi trường WINDOWS quen thuộc.



Hình 3.15. GNU Midnight Commander - The GNOME File Manager

3.3.6.2 Tìm file

Ngoài cây thư mục và các cửa sổ chính của trình quản lý File GNOME, thì trình quản lý File này còn cung cấp ở phía trên một hộp thoại (text box) 'Location' chỉ đường dẫn tới file (thư mục) được chọn. Bạn cũng có thể gõ vào hộp 'Location' này đường dẫn tới file (thư mục) cần chọn.

Bạn cũng có thể đưa vào hộp 'Location' này các địa chỉ FTP ('FTP sites') và trình quản lý File GNOME sẽ cố gắng kết nối tới địa chỉ ấy (tất nhiên trong trường hợp hệ thống của bạn phải kết nối mạng).

Các địa chỉ FTP phải được đưa vào theo dạng sau:

ftp://[địa chỉ trang Web]

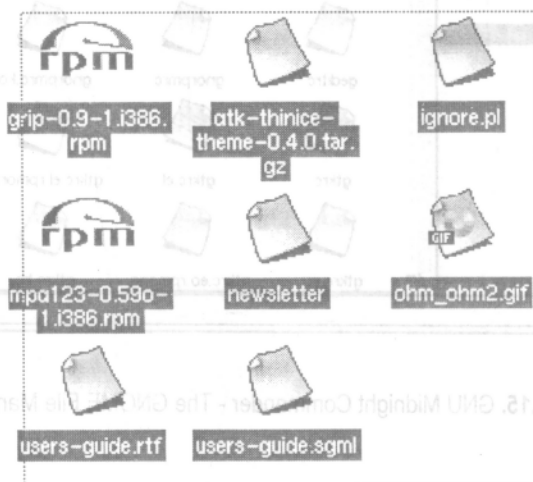
Ngoài ra, bạn có thể di chuyển giữa các thư mục, mở cửa sổ hiển thị nội dung của thư mục được chọn, ... giống như trong môi trường WINDOWS quen thuộc.

3.3.6.3 Chọn file

Một file sẽ được chọn bằng cách ấn chuột trái lên file ấy và file được chọn sẽ được hiện sáng.

Nếu muốn chọn một số file có vị trí liên tiếp nhau, bằng 2 cách sau:

- kích chuột lên file ở đầu (cuối), rồi ấn đồng thời phím 'Shift' và kích chuột lên file ở cuối (đầu). Vùng file được chọn sẽ hiện sáng.
- kích chuột lên file ở đầu (cuối), rồi ấn đồng thời phím 'Shift' trong khi kéo rê phím mũi tên xuống (mũi tên lên) tới vị trí của file cuối (đầu). Vùng file được chọn sẽ hiện sáng.



Hình 3.16. "Rubber Band Select"- Chọn file bằng cách bấm chuột và kéo.

Nếu các file cần chọn ở các vùng xa nhau (tức không có vị trí liên tiếp nhau), thì trong quá trình chọn các vùng file, bạn đồng thời có thể nhấn phím 'Ctrl' để vẫn giữ lại được các vùng file đã được chọn trước đó.

Bạn có thể chọn toàn bộ nội dung của một thư mục bằng cách chọn 'Select All' trong menu 'Edit'.

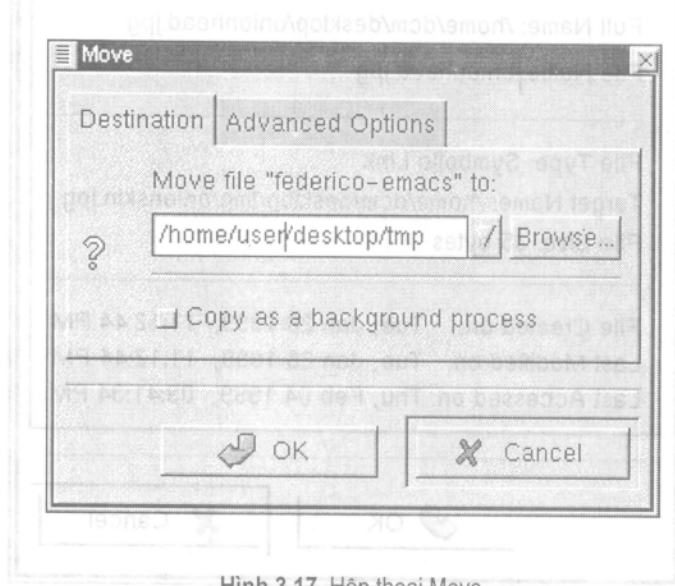
Bạn cũng có thể chọn các file bằng cách chọn 'Select Files' trong menu 'Edit'. Một hộp thoại sẽ hiện ra, bạn có thể đưa vào các tiêu chuẩn cho việc chọn file (thông qua tên của chúng). Ví dụ: phím * sẽ được hiểu là chọn tất cả các file

trong thư mục; D* sẽ được hiểu là chọn tất cả các file có tên bắt đầu bởi chữ 'D' có trong thư mục;

3.3.6.4 Sao chép và di chuyển các file

Trong trình quản lí File GNOME, và trong desktop, bạn có thể 'drag and drop' (kéo rê và thả) để di chuyển file. Bạn cũng có thể 'drag and drop' để sao chép một file, thư mục (các file, các thư mục) bằng cách nhấn nút 'Shift' trong quá trình kéo rê file, thư mục (các file, các thư mục) đó tới thư mục cần thiết. Tuy nhiên ta không thể di chuyển thư mục bằng cách 'drag and drop'.

Bạn còn có thể ấn nút chuột giữa lên file, thư mục được chọn để hiện ra menu, và bạn có thể chọn: 'Copy', 'Move', 'Link', hay 'Cancel Drag' đối với file, thư mục ấy.



Hình 3.17. Hộp thoại Move.

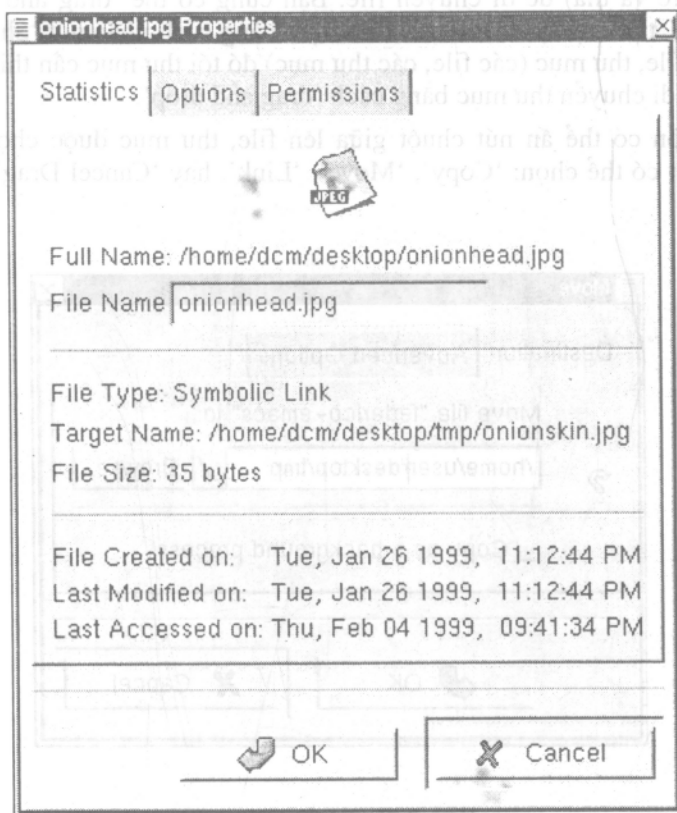
Ví dụ: trong hộp thoại 'Move', bạn hãy đưa đường dẫn tới vị trí đích của file đó. Tương tự với hộp thoại 'Copy' và 'Rename'

Trong trường hợp file được di chuyển là một liên kết ảo ('symbolic link'), bạn nên chọn 'Advanced Options' rồi chọn 'Preserve symlinks' để bảo toàn liên kết ảo.

3.3.6.5 Đổi tên file

Có 2 cách như sau để đổi tên file trong trình quản lí File GNOME, cũng như trong desktop:

- kích chuột phải lên file, rồi chọn 'Properties' trong menu được hiện ra. Rồi đưa tên mới của file trong hộp 'text box' 'Filename'.
- kích đúp chuột (kích chập) lên biểu tượng của file, sau đó đưa vào tên mới của file.



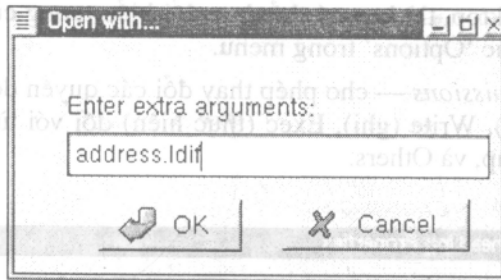
Hình 3.18. Hộp thoại File Properties

3.3.6.6 Chạy các ứng dụng từ trình quản lý File GNOME

Trong trình quản lý File GNOME, bạn có thể chạy một ứng dụng ngay từ cửa sổ chính bằng cách kích đúp chuột lên file được nối tới ứng dụng.

Nếu file đó không nối tới một ứng dụng, bạn có thể kích chuột phải lên file đó và chọn 'Open with' trong menu vừa xuất hiện. Ví dụ: nếu muốn soạn thảo file test.txt trong Emacs (một trình soạn thảo của Linux), hãy nhấn chuột phải lên

test.txt và chọn 'Open with'. Khi hộp thoại 'Open with' xuất hiện, hãy gõ vào 'emacs' rồi chọn 'OK', file test.txt sẽ được mở trong Emacs.



Hình 3.19. Hộp thoại Open With

3.3.6.7 Menu gắn với file

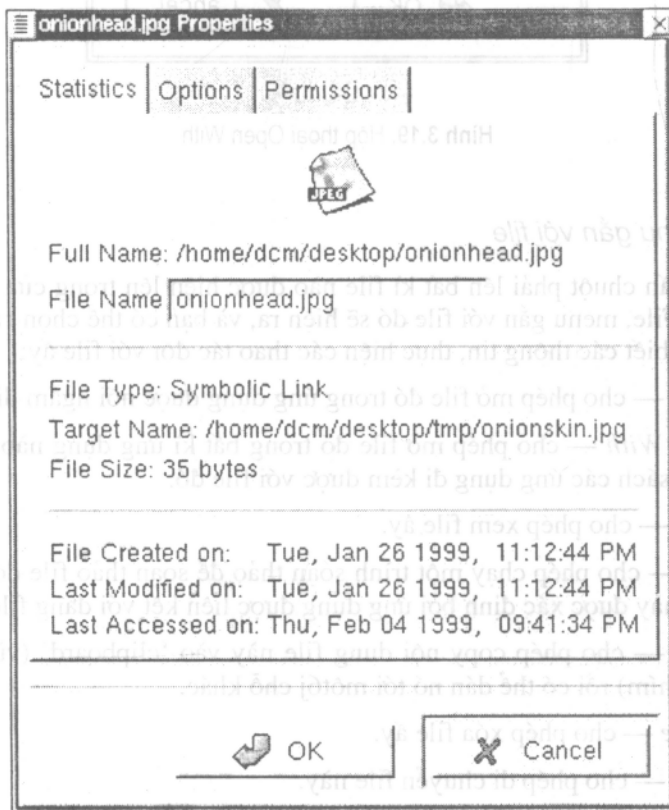
Khi nhấn chuột phải lên bất kỳ file nào được hiện lên trong cửa sổ chính của trình quản lý File, menu gắn với file đó sẽ hiện ra, và bạn có thể chọn các chức năng tương ứng để biết các thông tin, thực hiện các thao tác đối với file ấy:

- *Open* — cho phép mở file đó trong ứng dụng được nổi ngầm định với nó.
- *Open With* — cho phép mở file đó trong bất kỳ ứng dụng nào có tên trong danh sách các ứng dụng đi kèm được với file đó.
- *View* — cho phép xem file ấy.
- *Edit* — cho phép chạy một trình soạn thảo để soạn thảo file đó. Trình soạn thảo này được xác định bởi ứng dụng được liên kết với dạng file này.
- *Copy* — cho phép copy nội dung file này vào 'clipboard' (vùng đệm của bàn phím) rồi có thể dán nó tới một vị trí khác.
- *Delete* — cho phép xóa file ấy.
- *Move* — cho phép di chuyển file này.
- *Symlink* — cho phép tạo một liên kết ảo của file này tới một file khác mà bạn chọn (file này được đưa tên vào khi hộp thoại xuất hiện).
- *Properties* — làm xuất hiện hộp thoại 'Properties' cho phép bạn đưa vào cũng như xem các thuộc tính của file ấy. Hộp thoại 'Properties' có 3 mục chính: Statistics, Options, và Permissions.
 - ✓ *Statistics* — cho phép xem các thông tin của file gồm: name (tên file), type (kiểu file), size (kích thước), history. Bạn có thể thay tên của file bằng cách đưa tên mới vào trong hộp text 'File Name'.

- ✓ *Options* — cho phép thay đổi các hành động gắn với file này, như cách mở, xem, soạn thảo file đó.

Chú ý: nếu bạn làm hiện hộp thoại 'Properties' của một biểu tượng ở trên desktop, thì bạn có thể thay đổi biểu tượng của file đó bằng cách chọn mục 'Options' trong menu.

- ✓ *Permissions* — cho phép thay đổi các quyền đối với file như: Read (đọc), Write (ghi), Exec (thực hiện) đối với từng đối tượng: User, Group, và Others.



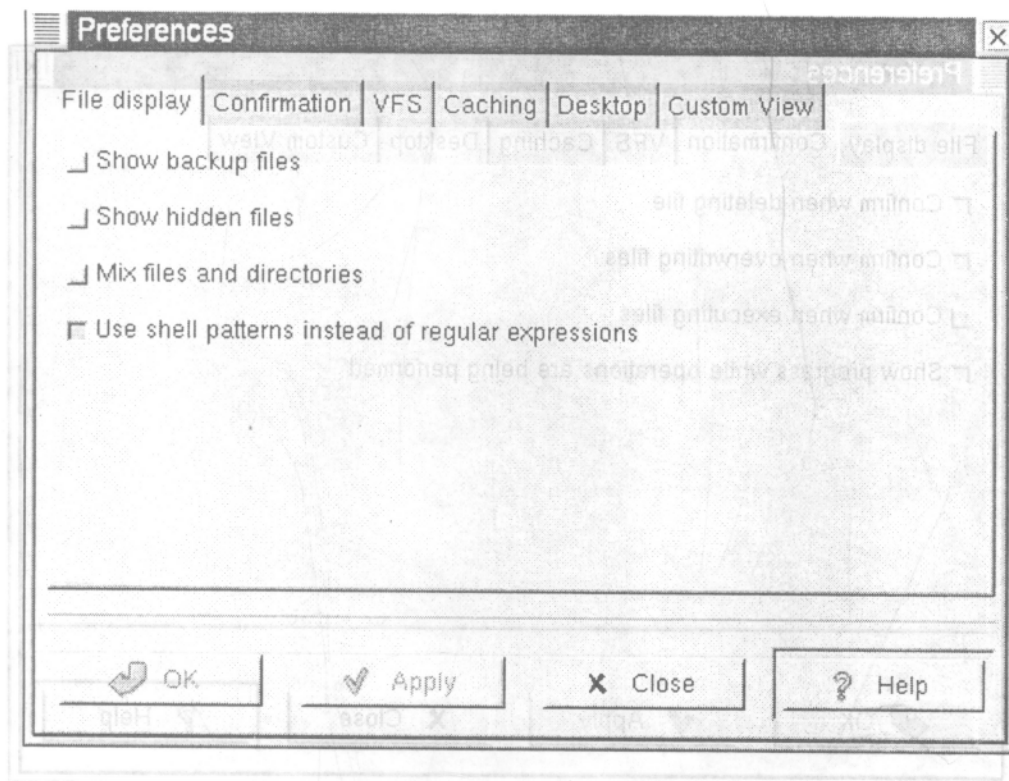
Hình 3.20. Hộp thoại File Properties

3.3.6.8 Thay đổi các tham chiếu trong trình quản lý File

Bạn có thể cấu hình trình quản lý File GNOME bằng cách chọn 'Preferences' trong menu 'Settings' để làm xuất hiện hộp thoại 'GMC Preferences'. Hộp thoại

'GMC Preferences' có 5 mục chính sau: File Display, Confirmation, VFS, Caching, Desktop, và Custom View.

File Display



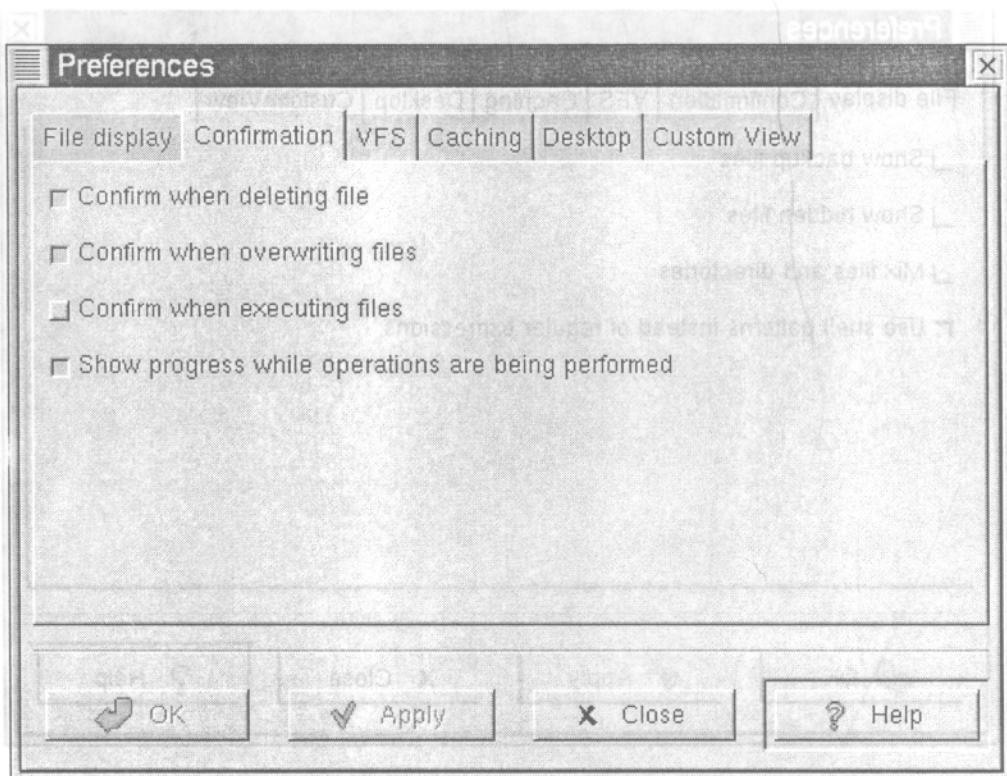
Hình 3.21. Hộp thoại Preferences của File Display

'File Display' cho phép bạn thay đổi cách hiển thị file trong GMC.

- *Show backup files* — cho phép nhìn được bất kỳ file 'backup' nào có thể có trong hệ thống.
- *Show hidden files* — cho phép nhìn được tất cả các file có tên bắt đầu bởi một dấu chấm ("dot files"). Các dạng file này thường là các file hay các thư mục cấu hình của hệ thống.
- *Mix files and directories* — cho phép hiển thị các file hay các thư mục theo trình tự mà bạn muốn sắp xếp chúng.

- *Use shell patterns instead of regular expressions* — cho phép bạn tạo các 'regular expressions' để sử dụng chúng trong sắp xếp và lọc file ('filters').

Confirmation



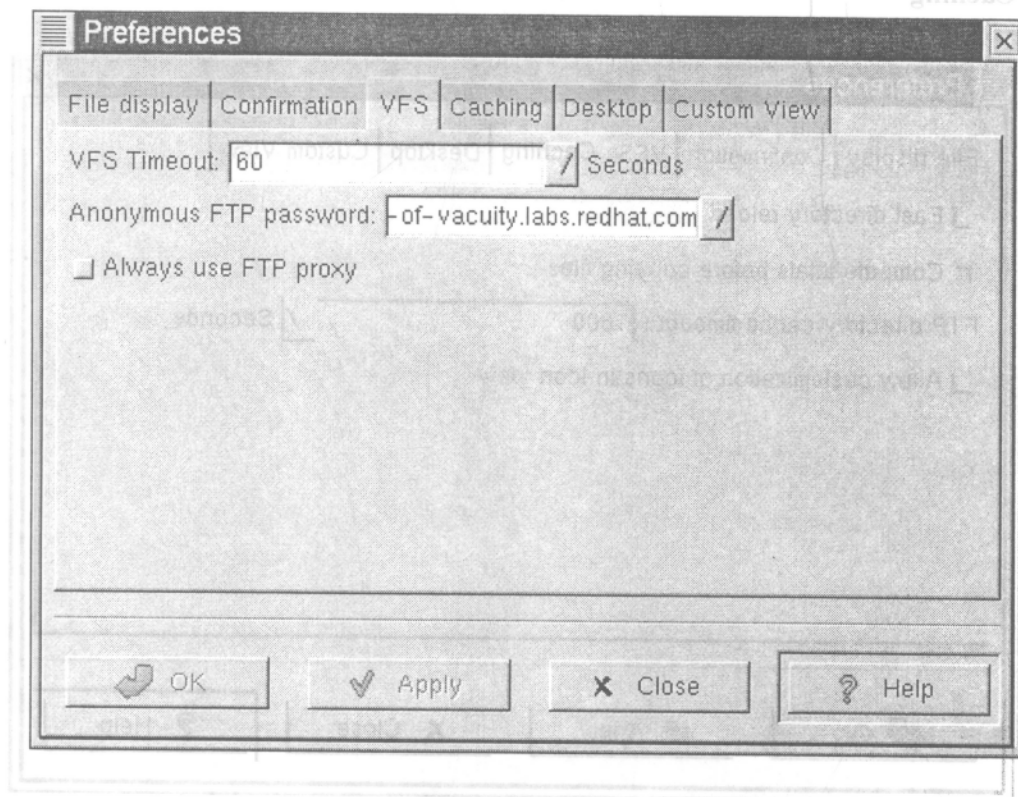
Hình 3.22. Hộp thoại Preferences của Confirmation

Mục này cho phép bạn thay đổi các hàm hỏi sự đồng ý của bạn trước khi tiếp tục chương trình ứng dụng.

- *Confirm when deleting file* — làm xuất hiện một hộp thoại đòi hỏi xác nhận trước khi xóa một file.
- *Confirm when overwriting files* - làm xuất hiện một hộp thoại đòi hỏi xác nhận trước khi ghi đè lên một file.
- *Confirm when executing files* — làm xuất hiện một hộp thoại đòi hỏi xác nhận trước khi chạy một file.

- *Show progress while operations are being performed* — làm hiện thanh tiến trình ('progress bar') trong quá trình thực hiện của một số thao tác như: sao chép, di chuyển, xóa, ...

VFS



Hình 3.23. Hộp thoại Preferences của VFS

Mục này cho phép bạn cấu hình các lựa chọn (options) của bạn trong hệ thống file ảo (Virtual File System-VFS).

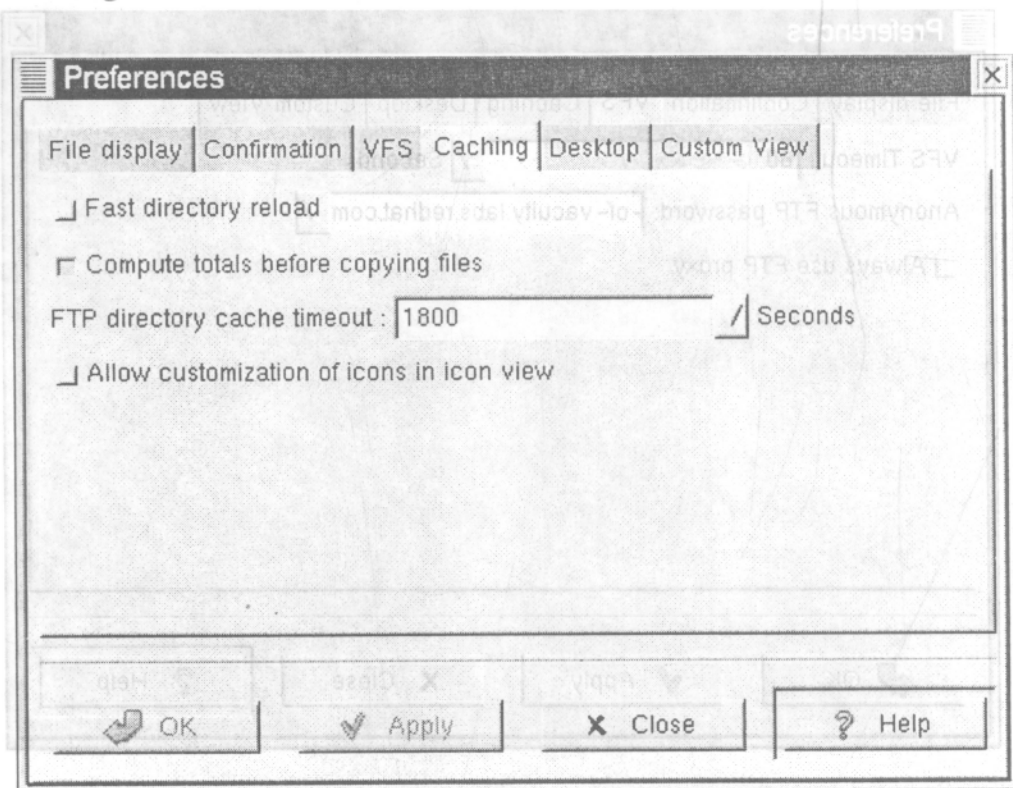
Hệ thống file ảo cho phép bạn điều khiển, làm việc với các file mà các file này không được định vị trên hệ thống của bạn.

- *VFS Timeout* — xác định khoảng thời gian chờ trong quá trình liên kết tới bất kỳ một VFS nào, khoảng thời gian này được tính bằng giây.
- *Anonymous FTP password* — cho phép đặt mật khẩu (password) khi xâm nhập (log into) vào các địa chỉ FTP vô danh (anonymous

FTP sites). Ví dụ bạn cần đặt mật khẩu cho địa chỉ email chẳng hạn.

- *Always use FTP proxy* — cho phép bạn chọn để sử dụng một proxy để liên kết tới các địa chỉ FPT (FTP sites).

Caching



Hình 3.24. Hộp thoại Preferences của Caching

Mục này cho phép bạn đặt các lựa chọn để tăng tốc độ xử lý của GMC bằng cách sử dụng bộ nhớ ẩn (cache).

- *Fast directory reload* — cho phép lưu thông tin về các thư mục trong bộ nhớ ẩn để việc tải (load) được nhanh hơn.

Chú ý: nếu chọn 'Fast directory reload', bạn có thể gặp một số vấn đề như: không nhìn được các file mới vừa được thêm vào trong các thư mục.

- *Compute totals before copying files* - GMC sẽ xác định số lượng các file mà bạn muốn sao chép trước khi quá trình sao chép được thực hiện.

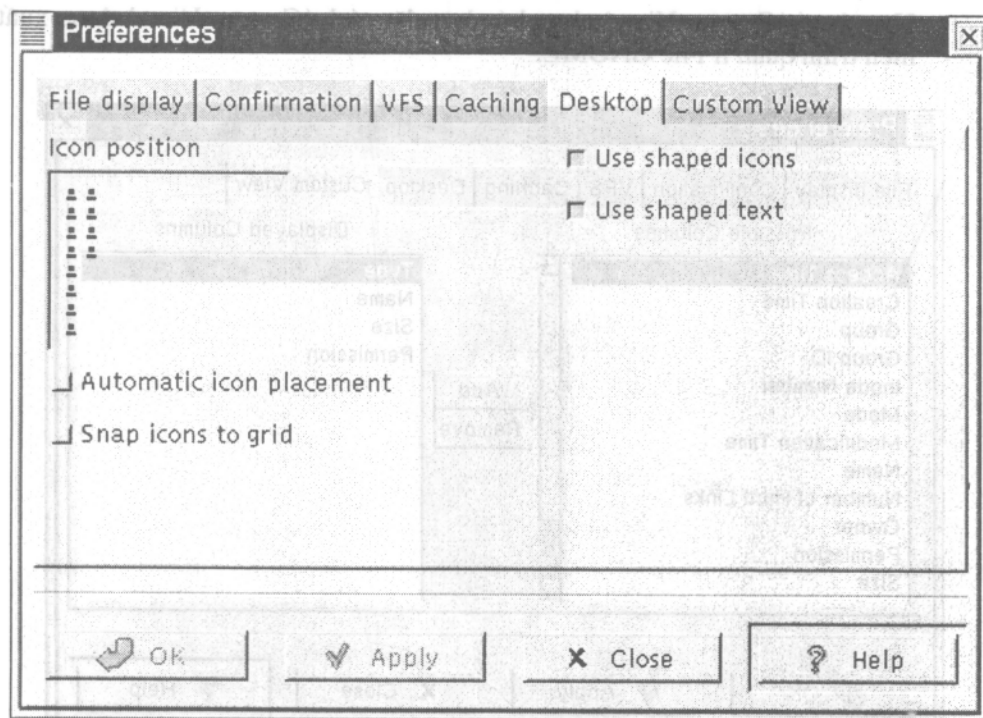
- *FTP directory cache timeout* — lựa chọn này sẽ cho phép giữ lại trong cache các thông tin về các địa chỉ FPT được thăm gần đây nhất ('recently visited FTP site') trong một khoảng thời gian do bạn chỉ định (theo giây).

- *Allow customization of icons in icon view* — cho phép bạn thay đổi biểu tượng trong 'icon view' bằng cách kích chuột phải lên chúng và chọn 'Properties' trong menu được xuất hiện.

Chú ý: việc chọn 'Allow customization of icons in icon view' có thể làm chậm tốc độ xử lý của hệ thống.

Desktop

Mục 'Desktop Dialog' sẽ cho bạn thay đổi các cài đặt của bạn đối với desktop.



Hình 3.25. Hộp thoại Desktop Dialog

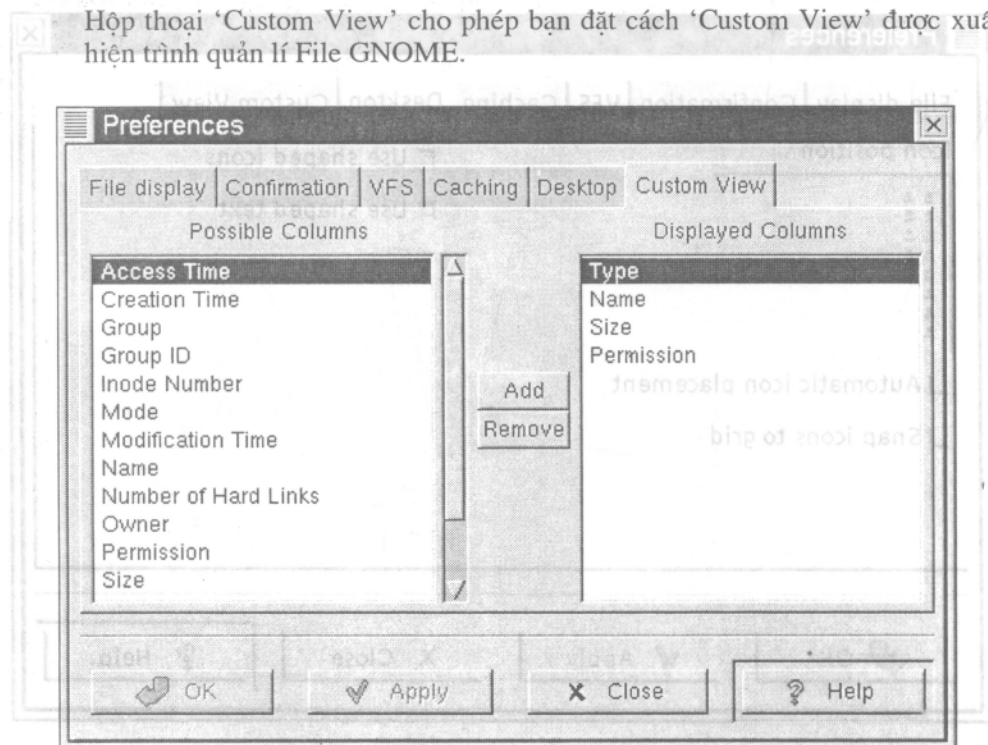
✓ *Icon placement* — cho phép bạn xác định vị trí mà các biểu tượng của desktop sẽ được đặt một cách ngẫu nhiên. Để thay đổi vị trí của các biểu

tượng này, hãy kích chuột lên chúng và kéo tới vị trí mà bạn muốn đặt biểu tượng ấy một cách ngẫu nhiên trên desktop.

- ✓ *Use shaped icons* — chức năng này được chọn theo ngẫu định. Do một số vấn đề có thể gặp phải đối với thiết bị hiển thị đồ họa,... nên việc chọn chức năng này sẽ đảm bảo cho tất cả các biểu tượng có trên desktop đều được nhìn thấy (non-transparent).
- ✓ *Use shaped text* — chức năng này được chọn theo ngẫu định. Bởi giống như 'Use shaped icons', nên việc chọn chức năng này sẽ đảm bảo cho tất cả các nhãn ('text labels') đi kèm với các biểu tượng có trên desktop đều được nhìn thấy.
- ✓ *Auto icon placement* — lựa chọn này sẽ cho phép việc sắp xếp tự động các biểu tượng có trên desktop. Nếu không chọn, bạn sẽ phải sắp xếp chúng bằng tay.
- ✓ *Snap icons to grid* — cho phép giữa nguyên toàn bộ các biểu tượng theo hàng lối ('in line') khi mà bạn di chuyển, kéo rê chúng.

Custom View

Hộp thoại 'Custom View' cho phép bạn đặt cách 'Custom View' được xuất hiện trình quản lý File GNOME.



Hình 3.26. Hộp thoại Custom View

Có 2 cột chính trong ‘Custom View’, đó là ‘Possible Columns’ và ‘Displayed Columns’.

- ‘Possible Columns’ chứa toàn bộ các dạng (types) thông tin có thể được hiển thị. Nếu bạn muốn thêm một trong số chúng, hãy chọn nút ‘Add’, và chúng sẽ được thêm vào trong danh sách ‘Displayed Columns’.
- Danh sách ‘Displayed Columns’ chỉ ra tất cả các dạng thông tin sẽ được đưa vào trong ‘Custom View’. Nếu như không thích một (một số) trong số chúng, bạn có thể bỏ bằng cách nhấn nút ‘Remove’.

Nếu như muốn sắp xếp lại vị trí của các nội dung trong danh sách ‘Displayed Columns’, bạn có thể dùng chuột để di chuyển nội dung đó tới vị trí mới mà bạn muốn.

3.3.6.9 Menu chính của trình quản lí File GNOME

Trong phần này, ta sẽ giới thiệu từng nội dung trong menu chính của trình quản lí File GNOME.

File

Menu này chứa các chức năng ở lớp trên được gắn với file.

- *Create New Window* — cho phép mở thêm một cửa sổ mới của trình quản lí File.
- *New / Terminal* — cho phép tạo thêm một cửa sổ ‘terminal’ của GNOME.
- *New / Directory* — cho phép tạo thêm một thư mục mới trong thư mục được hiển thị trong cửa sổ hiển thị chính (‘main window view’).
- *New / [application]* — cho phép đưa vào cho bạn các nội dung của ứng dụng trong menu ‘New’.
- *Open* — cho phép mở một file được chọn.
- *Copy* — làm xuất hiện hộp thoại ‘Copy’, và bạn có thể sao chép một file được chọn tới một vị trí khác.
- *Delete* — cho phép xóa file được chọn.
- *Move* — làm xuất hiện hộp thoại ‘Move/rename’, và bạn có thể thay đổi tên hoặc di chuyển file được chọn tới vị trí khác.
- *Show directory sizes* — xác định kích thước của các thư mục và hiển thị thông tin này trong cách hiển thị (view) ‘Detail’ hoặc ‘Custom’.
- *Close this Window* — cho phép đóng cửa sổ của trình quản lí File hiện tại.

Edit

Các nội dung trong menu này gắn với các chức năng liên quan tới việc soạn thảo và chọn file.

- *Select All* — cho phép chọn tất cả các file trong cửa sổ chính.
- *Select Files* — làm xuất hiện một hộp thoại, và bạn có thể đưa ra mẫu để chọn file ('selection pattern'). Ví dụ, để chọn tất cả các file có tên bắt đầu bởi chữ 'D', hãy gõ 'D*' rồi ấn nút 'OK'.
- *Invert Selection* — cho phép làm đảo việc chọn hiện tại. Ví dụ: bạn đã chọn 7 trong số tất cả 10 files có trong thư mục hiện thời, nếu muốn chọn 3 file còn lại thay vì chọn 7 file lúc đầu, bạn hãy chọn 'Invert Selection'.
- *Search* — cho phép tìm kiếm theo kiểu "completion" trong trình quản lí File.
- *Rescan Directory* — cho phép rà lại nội dung của thư mục hiện thời (trong trường hợp thư mục này vừa bị thay đổi).

Settings

Menu này chứa menu 'Preferences'.

- *Preferences* — sẽ làm xuất hiện hộp thoại 'Preferences', và bạn có thể thay đổi, cấu hình lại (customize) trình quản lí GNOME.

Layout

- *Sort By* — làm xuất hiện hộp thoại và phép bạn chọn cách sắp xếp các file có trong thư mục hiện thời. Việc sắp xếp có thể theo một trong các điều kiện sau:
 - Name (tên)
 - File Type (kiểu file)
 - Size (kích thước)
 - Time Last Accessed (lần truy nhập cuối cùng)
 - Time Last Modified (lần cập nhật)
 - Time Last Changed (lần thay đổi cuối cùng)
- *Filter View* — làm xuất hiện hộp thoại và phép bạn chỉ xem các file có tên trùng với mẫu ('names matching a pattern'). Ví dụ, để nhìn chỉ các file có tên bắt đầu bởi chữ 'D', hãy gõ 'D*' vào hộp thoại rồi ấn nút 'OK'. Nếu muốn nhìn tất cả các file, hãy mở lại hộp thoại 'Filter View', rồi gõ '*'.

- *Icon View* — cho phép hiển thị các file trong cửa sổ chính theo kiểu ‘large icons’.
- *Partial View* — cho phép hiển thị các file trong cửa sổ chính như một danh sách chỉ chứa tên của các file đó.
- *Full View* — cho phép hiển thị các file trong cửa sổ chính cùng với tất cả các thông tin về các file đó.
- *Custom View* — cho phép chuyển từ cửa sổ chính sang ‘Custom View’. Bạn có thể thay đổi các nội dung của ‘Custom view’ bằng cách sử dụng ‘Custom View Editor’ trong hộp thoại ‘Preferences’.

Commands

Menu này chứa các lệnh chạy trên các file trong trình quản lí File GNOME.

- *Find File* — làm xuất hiện hộp thoại cho phép bạn tìm các file trong hệ thống.
- *Edit mime types* — chức năng này sẽ chạy ‘GNOME Control Center Capplet’ và bạn có thể soạn (edit) ‘Mime Types’ của GNOME. ‘Mime types’ sẽ cùng với một số vấn đề khác, sẽ xác định xem ứng dụng nào sẽ kiểm soát các kiểu file nào.
- *Run Command* — cho phép bạn thực hiện một công lệnh từ GMC.
- *Run Command in panel* — cho phép bạn thực hiện các lệnh được tải trước (‘preloaded commands’) hoặc các công lệnh do bạn đưa ra trong thư mục hiện thời, như việc tìm SUID hay SGID của chương trình,....
- *Exit* — cho phép thoát hoàn toàn khỏi trình quản lí File GNOME.

3.3.6.10 ‘Drag and Drop’ trong GNOME

Chức năng ‘Drag and Drop’ trong GNOME được mở rộng và được ứng dụng một cách rất đa dạng, sau đây là một số ví dụ:

- *Drag a Color onto the Panel* — bạn có thể kéo một màu từ trong thanh chọn màu (‘selected color bar’) rồi thả nó vào thanh Panel, thanh Panel sẽ chuyển sang màu mà bạn vừa thả vào.
- *Drag a Pixmap to the Background Selector* — nếu muốn thay đổi màu nền bằng một ảnh, bạn có thể kéo ảnh đó từ trong trình quản lí File GNOME tới ‘Monitor Image’ trong mục ‘Background Capplet’ của ‘Control Center’, ảnh này sẽ được chuyển thành ảnh nền.
- *Drag to an Application* — một số ứng dụng trong GNOME cho phép việc ‘drag and drop’. Ví dụ: bạn có thể mở một URL bằng cách kéo rê URL này vào trong ‘Netscape 4x’, và ‘Netscape 4x’ sẽ mở URL này. điều này cho

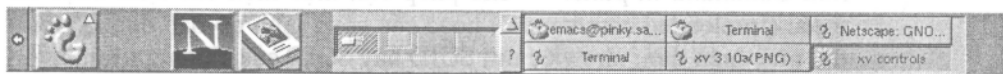
phép bạn mở một cách nhanh chóng một file nào đó khi làm việc trong trình quản lí File GNOME.

- *Adding an Application Launcher to the Panel* — cho phép bạn dùng chuột để kéo bất kì một file chạy được nào ('executable file') từ trình quản lí File GNOME, hoặc từ Desktop, và thả nó trên thanh Panel, để tạo một 'launcher' (bộ phóng) trên thanh Panel cho ứng dụng ấy.
- *Dragging Files* — bạn có thể kéo các file để sao chép, di chuyển chúng, tạo liên kết giữa các file, hoặc thả chúng từ cửa sổ của trình quản lí File lên desktop,....
- *Dragging Directories* — cho phép bạn kéo một thư mục ra khỏi trình quản lí File GNOME và thả chúng trên thanh Panel. Nó sẽ tạo ra một menu mới để cho phép bạn truy nhập dễ dàng hơn tới các file trong thư mục ấy.
- Bạn cũng có thể kéo bất kì một menu con nào từ menu chính (Main Menu) và thả trên thanh panel. điều này cho phép bạn truy nhập vào menu con đó một cách dễ dàng hơn.

3.3.7 Thanh Panel

3.3.7.1 Giới thiệu chung

Hình 3.27 minh họa hình ảnh của Panel trong GNOME.

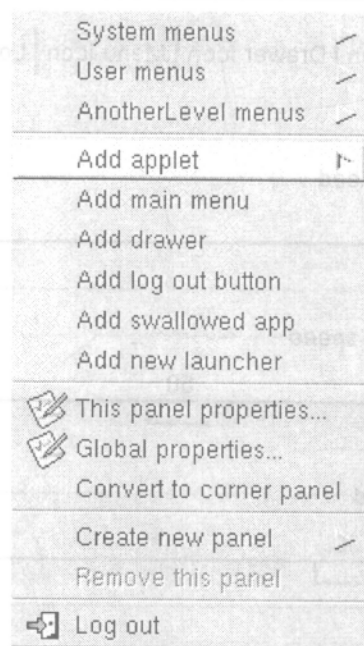


Hình 3.27. Thanh Panel ở dưới đáy màn hình

Từ Panel, bạn có thể chạy (một cách song song) các ứng dụng, các applets, các ứng dụng nhỏ, các 'utilities' (các biểu tượng của chúng nằm trên thanh Panel).

Sau đây là 2 chức năng chính:

- Thêm một 'applet': bạn hãy kích chuột phải trên thanh Panel như trong hình 3.28. Sau đó, hãy chọn một trong các applet được liệt kê trong submenu (menu con), biểu tượng của applet này sẽ xuất hiện trên thanh Panel.
- Thay đổi Panel: bạn có thể tạo thêm các thanh Panel cho desktop, chuyển thanh Panel từ vị trí nằm ngang thành nằm dọc, hoặc ẩn nó vào trong góc của desktop cũng như kéo nó hiện lại trên màn hình.



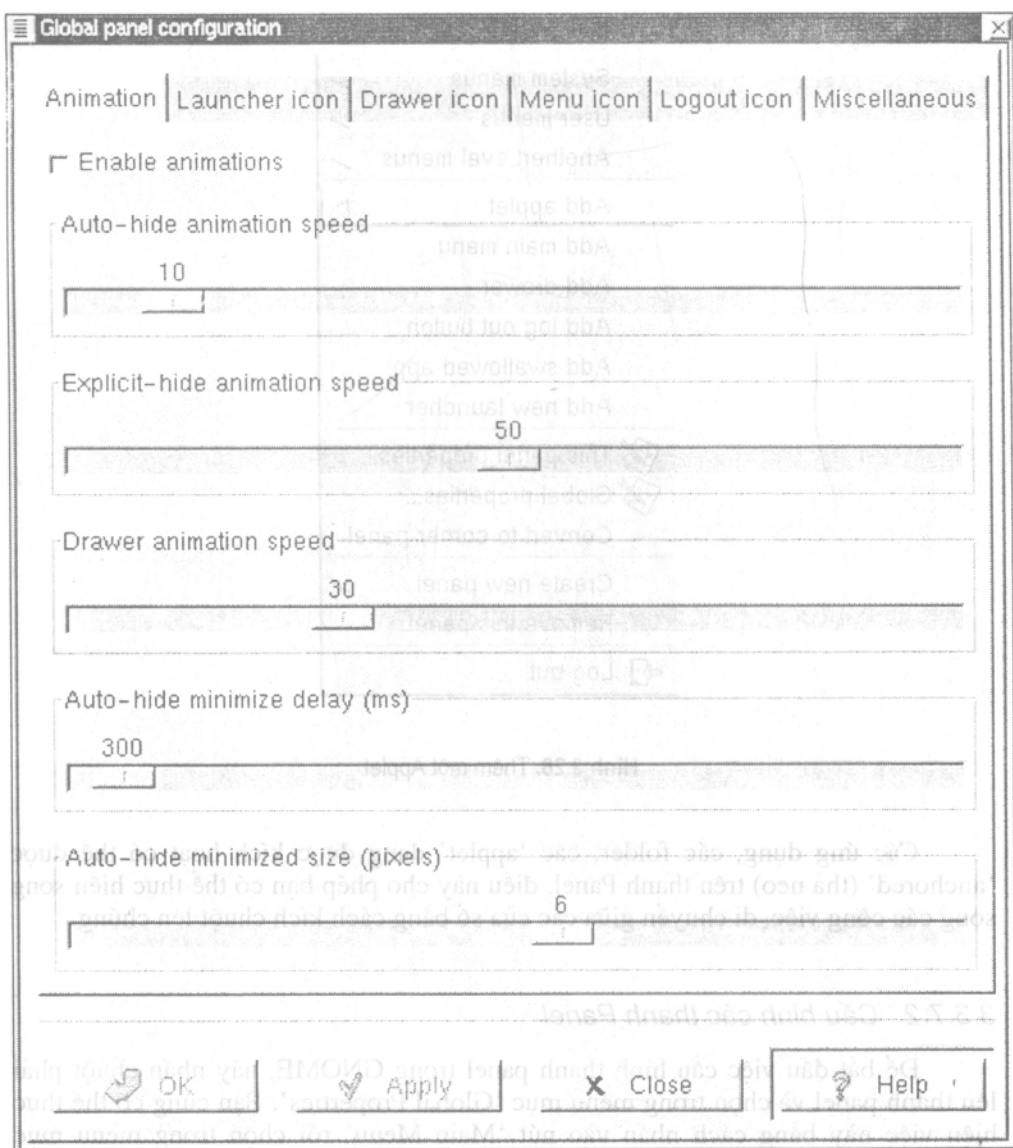
Hình 3.28. Thêm một Applet

Các ứng dụng, các folder, các 'applet' đang được kích hoạt có thể được 'anchored' (thả neo) trên thanh Panel, điều này cho phép bạn có thể thực hiện song song các công việc, di chuyển giữa các cửa sổ bằng cách kích chuột lên chúng.

3.3.7.2 Cấu hình các thanh Panel

Để bắt đầu việc cấu hình thanh panel trong GNOME, hãy nhấn chuột phải lên thanh panel và chọn trong menu mục 'Global Properties'. Bạn cũng có thể thực hiện việc này bằng cách nhấn vào nút 'Main Menu' rồi chọn trong menu mục 'Panel | Global Properties'.

Sau đó, hộp thoại 'Global Panel Configuration' sẽ xuất hiện (hình 3.29), và bạn có thể đặt các thuộc tính cho các thanh Panel hiện thời cũng như các Panel có thể được tạo ra sau này.



Hình 3.29. Hộp thoại Global Panel Configuration

Hộp thoại 'Global Panel Configuration' chứa 6 nội dung sau: Animation, Launcher Icon, Drawer Icon, Menu Icon, và Miscellaneous:

Animation

- *Enable animations*: mục này được chọn theo ngầm định.
- *Auto-Hide Animation Speed* — chọn này sẽ kiểm soát tốc độ xuất hiện lại của thanh Panel khi thanh Panel được đặt thuộc tính ẩn tự động.
- *Explicit-Hide Animation Speed* — chức năng này kiểm soát tốc độ ẩn khi bạn nhấn vào nút 'Hide' trên thanh Panel.
- *Drawer Animation Speed* — nếu bạn có sử dụng một panel 'Drawer', thuộc tính này sẽ kiểm soát sự xuất hiện nhanh chậm của menu 'Drawer'.
- *Auto-Hide Minimize Delay (ms)* — chức năng này sẽ kiểm soát khoảng thời gian trễ (mili giây) để thanh Panel bị thu nhỏ (trong trường hợp bạn cần thu nhỏ thanh Panel).
- *Auto-Hide Minimized Size (pixels)* — xác định tổng số điểm sẽ được nhìn thấy trong khi thanh Panel đã bị thu nhỏ (nếu thanh Panel được đặt thuộc tính tự động ẩn).

Launcher icon

- *Tiles enabled* — cho phép đặt nền đối với tất cả các biểu tượng nằm trên thanh Panel.
- *Tile filename (up)* — là tên và đường dẫn của file ảnh được chọn làm nền ở phía trên (không thể kích hoạt: inactive, not pressed).
- *Tile filename (down)* - là tên và đường dẫn của file ảnh được chọn làm nền ở phía dưới (có thể kích hoạt: active, pressed). Bạn có thể nhấn nút 'Browse' để tìm file ảnh này.
- *Border width (tile only)* — xác định độ rộng của đường viền xung quanh một 'icon' (biểu tượng).
- *Depth (displacement when pressed)*.

Drawer icon

- *Tiles enabled* — cho phép đặt nền đối với tất cả các 'drawers' nằm trên thanh Panel.
- *Tile filename (up)* — là tên và đường dẫn của file ảnh được chọn làm nền ở phía trên (không thể kích hoạt: inactive, not pressed). Bạn có thể nhấn nút 'Browse' để tìm file ảnh này.
- *Tile filename (down)* — là tên và đường dẫn của file ảnh được chọn làm nền ở phía dưới (có thể kích hoạt: active, pressed). Bạn có thể nhấn nút 'Browse' để tìm file ảnh này.
- *Border width (tile only)* — xác định độ rộng của đường viền xung quanh một 'tile'.
- *Depth (displacement when pressed)*.

Menu icon

- *Tiles enabled* — cho phép đặt nền cho nút ‘Main Menu’ trên thanh Panel.
- *Tile filename (up)* — là tên và đường dẫn của file ảnh được chọn làm nền ở phía trên (không thể kích hoạt: inactive, not pressed). Bạn có thể nhấn nút ‘Browse’ để tìm file ảnh này.
- *Tile filename (down)* — là tên và đường dẫn của file ảnh được chọn làm nền ở phía dưới (có thể kích hoạt: active, pressed). Bạn có thể nhấn nút ‘Browse’ để tìm file ảnh này.
- *Border width (tile only)* — xác định độ rộng của đường viền xung quanh một ‘tile’.
- *Depth (displacement when pressed)*.

Logout Icon

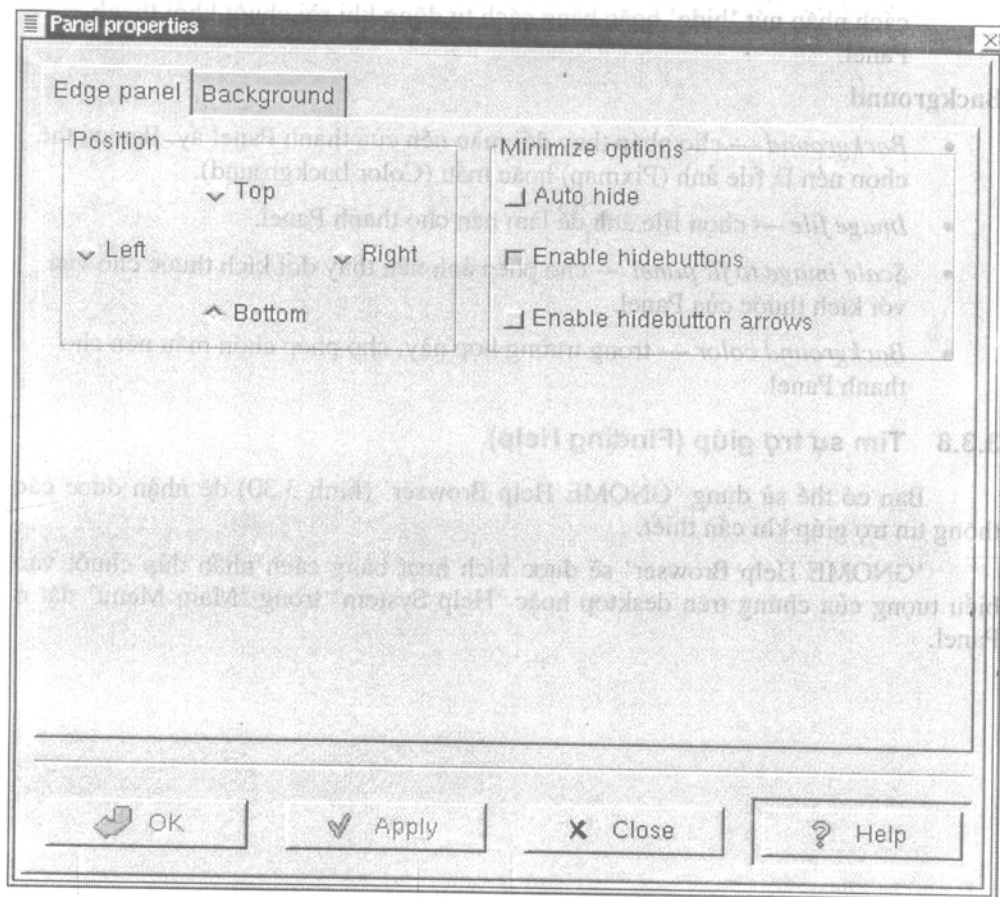
- *Tiles enabled* — cho phép đặt nền cho tất cả các nút ‘Logout’ trên thanh Panel.
- *Tile filename (up)* — là tên và đường dẫn của file ảnh được chọn làm nền ở phía trên (không thể kích hoạt: inactive, not pressed). Bạn có thể nhấn nút ‘Browse’ để tìm file ảnh này.
- *Tile filename (down)* — là tên và đường dẫn của file ảnh được chọn làm nền ở phía dưới (có thể kích hoạt: active, pressed). Bạn có thể nhấn nút ‘Browse’ để tìm file ảnh này.
- *Border width (tile only)* — xác định độ rộng của đường viền xung quanh một ‘tile’.
- *Depth (displacement when pressed)*.

Miscellaneous

- *Tooltips enabled* — cho phép hiển thị ‘tooltips’ (phân giải thích ngắn) đối với các nội dung trong thanh panel khi bạn di chuyển chuột tới nội dung ấy.
- *Show Small Icons* — cho phép hiển thị các biểu tượng con trong ‘Main Menu’.
- *Show ... buttons* — cho phép hiển thị các dấu chấm lửng trong các nội dung nằm trong ‘Main Menu’.
- *Show popup menus outside of panels.*
- *Keep menus in memory* — cho phép việc lưu các menu trong bộ nhớ mà không cần rà soát lại chúng. điều này làm tăng tốc độ của GNOME, nhưng nó có thể làm bỏ sót một số ‘item’ mới đã được thêm vào hệ thống.
- *Switched movement* — cho phép các biểu tượng và nút ‘launcher’ (phần text đi kèm với biểu tượng) sẽ di chuyển cùng với nhau.
- *Free Movement* (không ảnh hưởng tới các applet khác) — nhằm bảo tồn cách sắp xếp các ‘icon’ trên thanh Panel của bạn (trừ trường hợp bạn đang di chuyển chúng).

- *Prompt before logout* — nhằm hiển thị hộp thoại ‘yes/no’ trước khi bạn muốn ‘logout’ khỏi hệ thống.
- *Raise panels on mouse-over* — cho phép xuất hiện thanh Panel mỗi khi con trỏ chuột đang trỏ lên Panel.
- *Keep panel below windows* — nhằm cho phép các ứng dụng xuất hiện đè trên thanh Panel.
- *Close drawer if a launcher inside it is pressed* — lựa chọn này sẽ làm cho các ‘drawer’ tự động đóng lại khi bạn đã chọn xong một trong số các nội dung của chúng.
- *Applet padding* — cho phép thay đổi khoảng trống giữa các ‘icon’ và các ‘applet’.

3.3.7.3 Các đặc tính của mỗi Panel



Hình 3.30. Hộp thoại 'Panel properties'

Ta có thể cấu hình, một cách riêng lẻ, một thanh Panel đang được kích hoạt, bằng cách nhấn chuột phải lên thanh panel ấy, rồi chọn trong menu mục 'This panel properties'. Hoặc bạn có thể nhấn chuột vào nút 'Main Menu' và chọn mục 'Panel | This panel properties'.

Sau đó, hộp thoại 'Panel properties' sẽ xuất hiện, và ta có thể thay đổi các thuộc tính của thanh Panel đang được kích hoạt ấy.

Nội dung của hộp thoại này bao gồm: 'Edge Panel' và 'Background'. Trong trường hợp Panel đang được kích hoạt nằm trong góc ('corner panel'), thì 'Edge Panel' sẽ được thế bởi 'Corner Panel'.

Edge Panel

- *Position* — cho phép thay đổi vị trí của Panel trên màn hình.
- *Minimize Options* — cho phép ẩn thanh Panel, một cách tường minh, bằng cách nhấn nút 'hide' hoặc bằng cách tự động khi rời chuột khỏi thanh Panel.

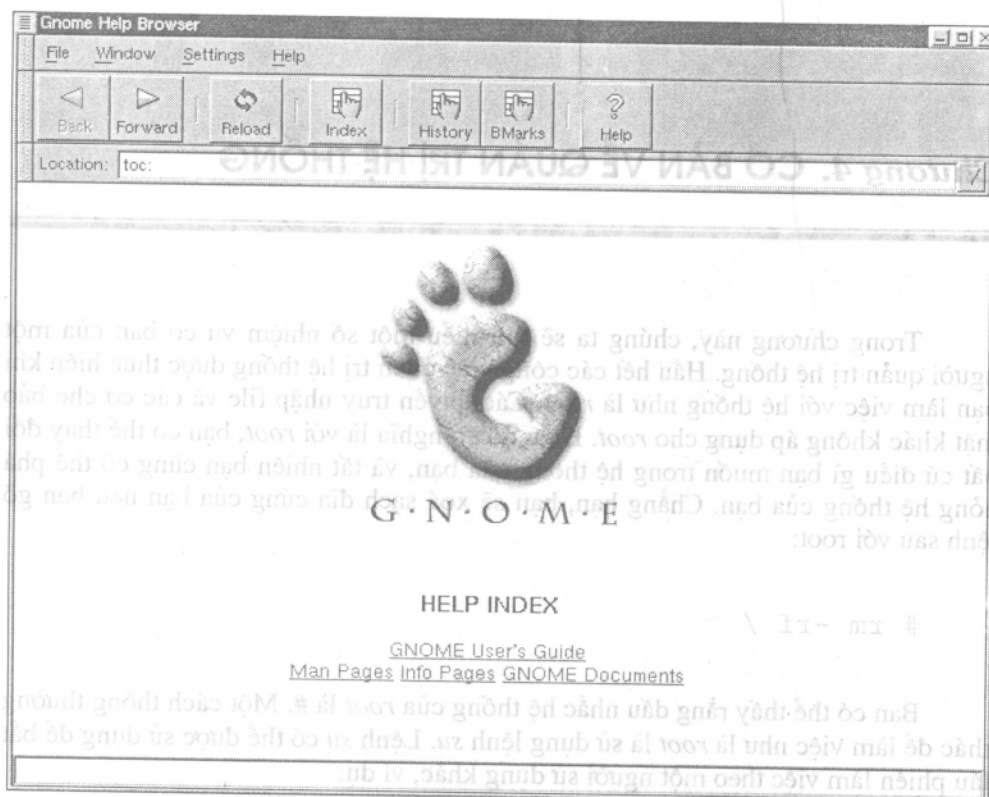
Background

- *Background* — cho phép thay đổi màu nền của thanh Panel ấy. Bạn có thể chọn nền là file ảnh (Pixmap) hoặc mẫu (Color background).
- *Image file* — chọn file ảnh để làm nền cho thanh Panel.
- *Scale image to fit panel* — cho phép ảnh nền thay đổi kích thước cho vừa với kích thước của Panel.
- *Background color* — trong trường hợp này, cho phép chọn mẫu nền cho thanh Panel.

3.3.8 Tìm sự trợ giúp (Finding Help)

Bạn có thể sử dụng 'GNOME Help Browser' (hình 3.30) để nhận được các thông tin trợ giúp khi cần thiết.

'GNOME Help Browser' sẽ được kích hoạt bằng cách nhấn đúp chuột vào biểu tượng của chúng trên desktop hoặc 'Help System' trong 'Main Menu' đặt ở Panel.



Hình 3.31. GNOME Help Browser

Chương 4. CƠ BẢN VỀ QUẢN TRỊ HỆ THỐNG

Trong chương này, chúng ta sẽ tìm hiểu một số nhiệm vụ cơ bản của một người quản trị hệ thống. Hầu hết các công việc quản trị hệ thống được thực hiện khi bạn làm việc với hệ thống như là *root*. Các quyền truy nhập file và các cơ chế bảo mật khác không áp dụng cho *root*. Điều đó có nghĩa là với *root*, bạn có thể thay đổi bất cứ điều gì bạn muốn trong hệ thống của bạn, và tất nhiên bạn cũng có thể phá hỏng hệ thống của bạn. Chẳng hạn, bạn sẽ xoá sạch đĩa cứng của bạn nếu bạn gõ lệnh sau với *root*:

```
# rm -rf /
```

Bạn có thể thấy rằng dấu nhắc hệ thống của *root* là #. Một cách thông thường khác để làm việc như là *root* là sử dụng lệnh *su*. Lệnh *su* có thể được sử dụng để bắt đầu phiên làm việc theo một người sử dụng khác, ví dụ:

```
$ su ducnh  
password: <gõ password tương ứng>
```

Lệnh *su* không có tham số sẽ cho phép bạn bắt đầu phiên làm việc với *root*. Bạn sẽ phải nhập mật khẩu của *root*. Một trong những nguyên tắc khi làm việc trên *root* là bạn phải luôn luôn kiểm tra kỹ lệnh trước khi thực hiện (ấn Enter). Ví dụ bạn muốn xoá thư mục, chẳng hạn, */usr/src/lodp* và vô tình bạn đã đánh một dấu cách sau dấu gạch chéo đầu tiên:

```
# rm -rf / usr/src/lodp
```

Lệnh này sẽ dẫn đến thảm họa. Một trong những lỗi hay gặp khác là nhầm lẫn các tham số của lệnh *dd*. Ví dụ, để ghi lại 1024 byte dữ liệu đầu tiên từ ổ đĩa */dev/hda* (nơi chứa bản ghi *boot* và bảng phân hoạch đĩa cứng), bạn có thể dùng lệnh:

```
# dd if=/dev/had of=/tmp/stuff bs=1k count=1
```

Tuy nhiên, nếu bạn tráo đổi *if* cho *of* trong câu lệnh trên thì các dữ liệu trên */tmp/stuff* (là nơi chứa các dữ liệu rác và lỗi) sẽ được ghi vào vị trí của bảng phân hoạch đĩa cứng!

4.1 KHỞI ĐỘNG VÀ THOÁT KHỎI HỆ THỐNG

4.1.1 Khởi động sử dụng đĩa mềm khởi động

Thông thường một đĩa mềm khởi động Linux chứa một ảnh của nhân hệ điều hành (kernel), ảnh này sẽ được đọc vào trong bộ nhớ khi khởi động bằng đĩa mềm. Người ta thường nén ảnh kernel bằng các thuật toán như trong chương trình nén *gzip*. Bạn có thể tạo ra đĩa mềm khi cài đặt Linux, hoặc bạn cũng có thể tự tạo ra đĩa mềm khởi động nếu có ảnh của kernel. Thông thường ảnh kernel đã có sẵn trên hệ thống và được lưu trong file */boot/vmlinuz* hoặc */boot/vmlinux*, hoặc */Image* trong một số bản Linux khác.

Giả sử bạn có ảnh kernel trong file */boot/vmlinuz*. Bây giờ để tạo ra một đĩa mềm khởi động, bước đầu tiên là bạn dùng lệnh *rdev* để đặt các *root device* (thiết bị root) vào file. Nếu bạn gõ lệnh:

```
# rdev /boot/vmlinuz
```

trên màn hình sẽ hiện ra thông báo về thiết bị mà hệ thống tìm thấy trong */boot/vmlinuz*:

```
courgette:/# rdev /boot/vmlinuz
Root device /dev/hda1
```

Và bây giờ bạn gõ lệnh:

```
courgette:/# rdev /boot/vmlinuz /dev/hda1
courgette:/#
```

Bây giờ để tạo một đĩa mềm khởi động, bạn phải có một đĩa mềm sạch (dùng lệnh *format* trên DOS hoặc bằng lệnh *fdformat* của Linux). Bạn sử dụng lệnh *dd* để copy ảnh kernel lên đĩa:

```
courgette:/# dd if=/boot/vmlinuz of=/dev/fd0 bs=8192
```

bs ở đây là kích thước khối (block size).

4.1.2 Khởi động sử dụng LILO

LILO (Linux LOader) là một chương trình nạp hệ điều hành đa năng, nó có thể nạp bất cứ hệ điều hành nào bạn cài trên máy tính, trong đó có Linux. Có rất nhiều cách để đặt cấu hình LILO. Chúng ta sẽ xét hai cách thông dụng nhất ở đây: cài đặt LILO trên *master boot record* (MBR) của đĩa cứng hoặc cài đặt LILO như một trình nạp hệ điều hành thứ hai chỉ dùng cho Linux.

Nếu bạn cài đặt LILO trên MBR của đĩa cứng, LILO sẽ cho phép bạn lựa chọn các hệ điều hành khác nhau mà bạn đã cài đặt trên máy để khởi động hệ thống, chẳng hạn DOS, Linux, Windows. Tuy nhiên, cả các hệ điều hành như OS/2 hay Windows NT đều có các trình nạp hệ điều hành riêng và cũng có thể cài lên MBR của đĩa cứng. Khi đó bạn có thể cài đặt LILO như trình nạp hệ điều hành thứ hai chỉ cho riêng Linux.

Nếu bạn cài đặt LILO sau hệ điều hành Windows, bạn chỉ việc đặt cấu hình cho LILO cho phép nạp hệ điều hành Windows giống như đối với hệ điều hành DOS. Tuy nhiên, nếu bạn cài đặt Windows sau khi bạn đã cài đặt LILO thì Windows sẽ ghi đè lên MBR của đĩa cứng. Điều này có nghĩa là bạn sẽ phải cài đặt lại LILO.

Để có thể đặt cấu hình của LILO, trước hết bạn kiểm tra cấu hình trong file */etc/lilo.conf*.

```
# more /etc/lilo.conf
```

Phần đầu tiên của file này thiết lập các tham số cơ bản:

```
boot = /dev/had
```

```
compact
install = /boot/boot.b
map = /boot/map
```

Dòng đầu tiên *boot = /dev/had* chỉ ra nơi LILO được cài đặt trên *boot record*. Nếu bạn có đĩa cứng SCSI, bạn sử dụng tên thiết bị là: *dev/sha*. Nếu bạn chỉ ra một phân đoạn thiết bị thay vì thiết bị, chẳng hạn */dev/hda2*, LILO sẽ được cài đặt như là trình nạp hệ điều hành thứ hai trên phân đoạn này.

Dòng *compact* có nghĩa là Linux sẽ thực hiện một số tối ưu hoá. Các dòng lệnh *install* và *map* nên được giữ nguyên, chúng chỉ ra file sẽ chứa *boot sector* để sử dụng trên MBR và file *map*.

Bây giờ đối với mỗi hệ điều hành mà bạn muốn LILO nạp vào, bạn hãy thêm một đoạn mã vào file */etc/lilo.conf*. Ví dụ đoạn mã cho Linux như sau:

```
image = /boot/vmlinuz # vị trí của kernel
label = linux          # tên hệ điều hành
root = /dev/hda2       # vị trí bảng phân hoạch root
vga = ask              # hỏi chế độ văn bản VGA khi khởi động
```

Nếu bạn có nhiều kernel Linux, bạn có thể lựa chọn một trong các kernel của bạn để khởi động. Bạn chỉ việc thêm một đoạn mã *image* cho mỗi kernel. Trường *label* là trường bắt buộc phải có. Nếu bạn không xác định các trường *root* và *vga* thì bạn có thể xác định các trường này bằng lệnh *rdev*.

Đoạn mã cho MS-DOS và Windows 95/98 sẽ như sau:

```
other = /dev/hda1      # vị trí bảng phân đoạn
table = /dev/had       # vị trí bảng phân đoạn cho /dev/hda2
label = linux          # tên hệ điều hành
```

Nếu bạn muốn khởi động bằng phân đoạn MS-DOS hay Windows 95/98 nằm trên ổ đĩa thứ hai, bạn thêm vào dòng lệnh:

```
loader = /boot/any_d.b
```

Khi bạn đã chuẩn bị xong file */etc/lilo.conf*, bạn có thể chạy lệnh:

```
# /sbin/lilo
```

Lệnh này sẽ làm việc như sau:

```
courgette:/# /sbin/lilo
Added linux
Added msdos
courgette:/#
```

Bây giờ bạn có thể khởi động lại hệ thống bằng lệnh *reboot*. Hệ điều hành đầu tiên trong danh sách sẽ được nạp một cách mặc định. Để có thể lựa chọn các hệ điều hành khác nhau, bạn chỉ cần giữ phím *Shift* hoặc *Ctrl* hay đơn giản chỉ cần ấn phím *Scroll Lock* khi khởi động. Khi đó LILO sẽ hiện lên dấu mời:

```
boot:
```

Bạn có thể ấn phím Tab để lựa chọn từ danh sách các hệ điều hành sẵn có:

```
boot: <Tab>
linux msdos
```

Sử dụng LILO như một trình nạp hệ điều hành thứ hai

Nếu trên hệ thống của bạn đã sẵn có trình nạp hệ điều hành của OS/2 hoặc Windows NT, bạn có thể đặt LILO như một trình nạp hệ điều hành thứ hai bằng cách thay đổi dòng lệnh *boot=...* trong file */etc/lilo.conf* thành tên của bảng phân hoạch Linux, ví dụ:

```
boot = /dev/hda2
```

Khi đó LILO sẽ được cài đặt trên *boot record* của */dev/hda2*.

Gỡ bỏ LILO

Nếu bạn muốn gỡ bỏ LILO đã cài đặt trên MBR, cách dễ nhất là sử dụng lệnh FDISK của MS-DOS:

```
FDISK /MBR
```

Lệnh này sẽ thay thế MBR bằng một *boot record* của MS-DOS mới.

LILO ghi lại bản sao của *boot record* trong file */boot/boot.0300* (cho các ổ IDE) và */boot/boot.0800* (cho các ổ SCSI). Bạn có thể sử dụng lệnh *dd* để thay thế *boot record* trên ổ đĩa với bản sao này. Ví dụ:

```
# dd if=/boot/boot.0300 of=/dev/had bs=446 count=1
```

Lệnh này sẽ copy 446 byte từ file */boot/boot.0300* vào */dev/had*.

4.1.3 Thoát khỏi hệ thống

Cách dễ nhất để thoát khỏi hệ thống Linux là sử dụng lệnh *shutdown*. Khuôn dạng chung của lệnh như sau:

```
# Shutdown time warning-message
```

trong đó *time* là thời gian thoát khỏi hệ thống và *warning-message* là thông báo cho mọi người sử dụng trên hệ thống biết trước khi thoát khỏi hệ thống. Một cách khác, bạn có thể đặt tham số *time* là *now* để tắt khỏi hệ thống tức thời. Nếu bạn thêm lựa chọn *-r* thì hệ thống sẽ khởi động lại sau khi thoát.

Ví dụ, để thoát khỏi hệ thống và khởi động lại vào lúc 8h:00 tối, bạn sử dụng lệnh:

```
# shutdown -r 20:00
```

Lệnh *halt* cũng có thể sử dụng để thoát khỏi hệ thống một cách tức thời không có thông báo. Sau khi hiện ra dòng thông báo:

```
The system is halted.
```

Bạn mới có thể có thể tắt máy. Trên một số hệ thống, bạn còn có thể dùng tổ hợp phím Ctrl-Alt-Del để thoát khỏi hệ thống thay cho lệnh *shutdown*.

4.1.4 File */etc/inittab*

Khi khởi tạo hệ thống, công việc đầu tiên là khởi tạo kernel. Trong quá trình này, kernel sẽ tìm kiếm tất cả các thiết bị có thể và khởi tạo chúng. Sau đó, kernel thực hiện chương trình *init* ở trong */etc*, */bin* hoặc */sbin*. Đây là một chương trình đa năng sinh ra các tiến trình mới và khởi động lại một số chương trình khi chúng kết thúc. Ví dụ mỗi một *console* ảo có một tiến trình *getty* chạy trên nó, tiến trình này được *init* khởi tạo. Khi bạn thoát khỏi phiên làm việc trên một *console* ảo, tiến trình *getty* dừng và *init* sẽ khởi tạo lại một tiến trình khác, cho phép bạn có thể *login* lại.

Sau khi khởi động, chương trình *init* hoạt động trên nền hệ thống. Các cấu hình cho chương trình này được đặt trong file */etc/inittab*. Một ví dụ về file này như sau:

```
#
# inittab          This file describes how the INIT
process should set up
#                  the system in a certain run-level.
#
# Author:          Miquel van Smoorenburg,
<miquels@drinkel.nl.mugnet.org>
#                  Modified for RHS Linux by Marc Ewing
and Donnie Barnes
#

# Default runlevel. The runlevels used by RHS are:
#  0 - halt (Do NOT set initdefault to this)
#  1 - Single user mode
#  2 - Multiuser, without NFS (The same as 3, if you
do not have networking)
#  3 - Full multiuser mode
#  4 - unused
#  5 - X11
#  6 - reboot (Do NOT set initdefault to this)
```

```
#
id:3:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6

# Things to run in every runlevel.
ud::once:/sbin/update

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we
have a few minutes
# of power left.  Schedule a shutdown for 2 minutes
from now.
# This does, of course, assume you have powerd
installed and your
# UPS connected and working correctly.
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure;
System Shutting Down"

# If power was restored before the shutdown kicked in,
cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored;
Shutdown Cancelled"
```



```
# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6

# Run xdm in runlevel 5
# xdm is now a separate service
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Khi hệ thống khởi động, file */etc/inittab* sẽ khởi động 6 *console* ảo, mỗi *console* ảo được gắn với một tiến trình *tty* tương ứng. Các bạn có thể nhìn thấy các *run levels* trong file */etc/inittab*. Chương trình *init* sẽ duyệt qua chuỗi các *run levels* này tương ứng với các trạng thái hoạt động của hệ thống. Hệ thống sẽ vào *run level 1* ngay sau khi khởi động. Các *run levels 2* và *3* là các trạng thái thông thường đa người dùng của hệ thống. *Run levels 4* sẽ khởi động hệ thống X Window bằng chương trình quản trị màn hình X *xdm*. *Run levels 6* sẽ khởi động lại hệ thống.

Ví dụ, lệnh sau sẽ duy trì dấu mời *login* cho một *terminal* cho các *run levels* từ 1 đến 5:

```
s2:12345:respawn:/sbin/agetty -L 9600 ttyS1 vt100
```

trong đó :

- "s2" là tên được sử dụng bởi chương trình *init*.
- *respawn* là một từ khoá của *init* thường được sử dụng cho các *terminal*. Nếu sau một khoảng thời gian, chương trình *agetty* không nhận được trả lời từ phía người sử dụng thì chương trình sẽ kết thúc. "respawn" sẽ cho *init* biết khởi động lại *agetty*, đảm bảo rằng luôn luôn có dấu mời *login*: trên *terminal* cho dù người sử dụng có vào hay không.

Để biết rõ hơn về các chương trình *init* và *agetty*, bạn có thể xem trong các trang hướng dẫn *man*.

4.2 QUẢN TRỊ HỆ THỐNG FILE

Một trong các nhiệm vụ quan trọng nhất của quản trị viên hệ thống là quản lý đĩa cứng và hệ thống file của Linux. Nhiệm vụ này bao gồm việc kiểm tra định kỳ hệ thống file. Các công việc quản trị hệ thống file thường gặp là :

- Tìm các sector hỏng trong hệ thống file.
- Kiểm tra tính chặt chẽ và tính đúng đắn của bảng i-node của hệ thống file.
- Kiểm tra các quyền truy nhập và sở hữu để đảm bảo các truy nhập là thích đáng.
- Làm cho các hệ thống file (địa phương hoặc ở xa) sẵn sàng cho người sử dụng.
- Quản lý không gian đĩa của hệ thống Linux.
- Thường xuyên sao lưu dữ liệu để đảm bảo an toàn.

Mặc dù một vài trong số các hành động kể trên thực hiện tự động khi khởi động Linux (như kiểm tra các sector hỏng), ta cũng nên biết cách thực hiện các thủ tục này bằng tay, cũng như cách chúng thực hiện để sửa chữa các lỗi có thể xảy ra.

4.2.1 Gắn kết (mount) và tháo bỏ (unmount) hệ thống file

Trong Linux khái niệm hệ thống file để chỉ một thiết bị ngoại vi được định dạng để có thể lưu các file cho phép truy cập ngẫu nhiên ví dụ đĩa cứng, đĩa mềm, CD-ROM ... (bảng từ chỉ cho phép truy nhập tuần tự nên không thể chứa một hệ thống file)

Nguyên nhân của cơ chế gắn kết các hệ thống file là cách Linux tổ chức đĩa và các hệ thống file tạo nên toàn bộ cấu trúc thư mục. Linux sử dụng một cấu trúc thư mục duy nhất, không quan tâm đến việc lưu trữ trên bao nhiêu đĩa và bao nhiêu phân hoạch. Mỗi hệ thống file của mỗi phân hoạch phải là một phần của cấu trúc thư mục lớn hơn. Toàn bộ cây thư mục chỉ có một thư mục gốc, các hệ thống file khác được nối vào ở các mức thấp hơn.

Để có thể hình dung khái niệm này, hãy tưởng tượng một hệ thống file Linux chuẩn với một phân hoạch gốc (/) ở trên cùng, các phân hoạch khác được chia nhánh từ phân hoạch gốc. Phân hoạch gốc nằm trên một phân hoạch của đĩa cứng đầu tiên. Thông thường, đĩa đó có cả các thư mục khác trên nó như /dev, /lib, /etc, ... tất cả các thư mục cần thiết cho việc khởi động một hệ điều hành Linux nhỏ nhất phải được đặt trên phân hoạch gốc.

Tuy vậy giả sử rằng bạn muốn có một hệ thống file /usr rất lớn để hỗ trợ nhiều người dùng với các cơ sở dữ liệu file rất lớn. Phân hoạch gốc có thể không có đủ chỗ cho tất cả các file mà ta muốn lưu, khi đó bạn có thể dùng một phân hoạch khác (trên cùng hoặc trên một đĩa cứng khác), định dạng nó thành một hệ thống file của Linux, và sau đó nối chúng vào hệ thống file gốc dưới thư mục /usr. Mỗi khi người dùng di chuyển từ thư mục /bin (trong phân hoạch thứ nhất) đến /usr/tparker chẳng hạn, người dùng đã di chuyển đến một phân hoạch hay đĩa khác. Người dùng không thể nhận thấy được sự di chuyển giữa các phân hoạch vì các phân hoạch trông giống như một cây thư mục duy nhất. Thư mục /usr được gọi là được gắn kết vào thư mục gốc.

Một cách chính xác hơn phân hoạch chứa hệ thống file /usr được nối vào hệ thống file gốc tại vị trí thư mục /usr. Nó cũng có thể được nối vào thư mục /home, Linux không quan tâm đến việc ta nối hệ thống file vào đâu, miễn là nó được nối vào một thư mục tồn tại trong hệ thống file gốc và còn trống, nếu không nội dung của thư mục sẽ bị xóa.

Ta cũng có thể nối một hệ thống file vào một hệ thống file khác đã được nối với hệ thống file gốc. Ví dụ nối một CD-ROM vào thư mục /usr/tparker/cd_rom, trong đó /usr là một hệ thống file mà ta đã nối vào hệ thống file gốc. Một lần nữa sự di chuyển giữa các hệ thống file là trong suốt với người dùng.

Linux cho phép ta gắn kết các phân hoạch vào bất kì chỗ nào, từ của bất kì nguồn nào, miễn là chúng phù hợp với toàn bộ cấu trúc hệ thống file. Chỉ có một chỗ không thể nối các hệ thống file là thư mục gốc nằm trong hệ thống file gốc. Linux cũng cho phép gắn kết một số hệ thống file của các hệ điều hành khác như DOS hay OS/2 vào hệ thống file của Linux. Để thực hiện điều đó Linux cần biết chỗ truy nhập hệ thống file đó (thư mục /dos, /usr/dos, hoặc một số thư mục khác) và kiểu của hệ thống file đó. Sau khi đã gắn kết Linux sẽ cho phép người sử dụng di chuyển giữa các thư mục và file của hệ thống đó như trong bất kì thư mục nào khác của Linux. Tuy vậy, tại một thời điểm, một hệ thống file chỉ có thể được gắn kết vào một vị trí, chẳng hạn ta không thể nối một hệ thống file dưới cả /usr và /home.

Khả năng gắn kết các hệ thống file làm cho Linux trở nên linh hoạt. Nếu một muốn nối một ổ cứng chứa dữ liệu là hệ thống file mà Linux có thể hiểu được, người sử dụng có thể cắm ổ cứng này vào máy, sau đó gắn kết hệ thống file của nó với bất kì chỗ nào trên hệ thống file của Linux, và như vậy người sử dụng có thể truy nhập được dữ liệu trên ổ cứng đó. Mọi thiết bị có thể chứa hệ thống file đều có thể được nối: CD-ROM, đĩa mềm, đĩa quang, ...

Linux cung cấp lệnh mount để gắn kết một hệ thống file. Cú pháp câu lệnh như sau:

```
# mount device_name mount_point
```

trong đó `device_name` là tên thiết bị lưu trữ (phân hoạch, đĩa cứng, CD-ROM,...), `mount_point` là tên thư mục mà thiết bị được gắn kết vào. Ví dụ câu lệnh để nối `/dev/sda4` (phân hoạch thứ 4 của ổ đĩa SCSI thứ nhất) vào thư mục `/usr` là:

```
# mount /dev/sda4 /usr
```

Để nối một hệ thống file CD-ROM (như `/dev/cdrom` vào thư mục `/cdrom` (giả sử đã tồn tại), ta dùng lệnh :

```
# mount /dev/cdrom /cdrom
```

Lưu ý là chỉ có root mới đủ quyền thực hiện công việc này. Mặc dù có thể cho phép người dùng nối các hệ thống file nhưng điều này có thể dẫn đến các vấn đề an toàn, do đó không được khuyến khích.

Ta có thể nối một hệ thống file theo kiểu chỉ đọc, đặc điểm này cho phép cấm ghi lên hệ thống file được nối (có thể chứa những dữ liệu mà ta không muốn làm hỏng), mọi cố gắng ghi lên nó sẽ sinh ra thông báo lỗi. Để nối một hệ thống file theo kiểu chỉ đọc, dùng lựa chọn `-r` :

```
# mount -r /dev/cdrom /cdrom
```

hoặc như một số phiên bản cũ của UNIX và Linux cho phép viết `-r` ở cuối

```
# mount /dev/cdrom /cdrom -r
```

Khi một hệ thống file không còn được gắn kết trong hệ thống file của Linux nữa (do đó người dùng không thể truy nhập đến các thư mục nơi nối nó), hệ thống file được gọi là đã bị tháo bỏ. Tất cả các hệ thống file đều có thể bị tháo bỏ trừ hệ thống file gốc. Để tháo bỏ một hệ thống file, người sử dụng dùng lệnh `umount`. Lệnh `umount` lấy tên của thiết bị hoặc điểm nối làm tham số. Để bỏ nối CD-Rom vừa được nối trong ví dụ trên, ta dùng một trong hai lệnh sau :

```
# umount /dev/cdrom
# umount /cdrom
```

Ta không cần bỏ nối cho mọi hệ thống file trước khi đóng hệ thống, Linux sẽ tự động thực hiện công việc đó trong quá trình đóng hệ thống.

4.2.2 Gắn kết các hệ thống file tự động bằng file `/etc/fstab`

Các hệ thống file đã được nối từ trước đều không cần nối tự động khi hệ thống khởi động lại (khác với root, luôn được nối tự động khi khởi động hệ thống). Khi khởi động Linux, cần phải biết tìm các hệ thống file ở đâu để gắn kết. Linux dùng file `/etc/rc` (chạy khi Linux khởi động) để thực hiện lệnh :

```
# mount -av
```

Khi thực hiện lệnh này, Linux biết được phải đọc file `/etc/fstab` để tìm ra hệ thống file cần gắn kết cũng như nơi cần gắn kết.

Linux cũng cung cấp lệnh *mountall* để nối tất cả các hệ thống file trong file */etc/fstab*, tuy vậy không phải mọi phiên bản của Linux đều hỗ trợ câu lệnh này.

Mỗi dòng trong file */etc/fstab* (là file ASCII) có khuôn dạng như sau :

```
device mount_location filesystem_type options
dump_frequency pass_number
```

trong đó : *device* là tên thiết bị lưu trữ, tiếp theo là điểm nối, kiểu hệ thống file, tiếp theo là các chỉ dẫn làm thế nào để xử lý hệ thống file.

Sau đây là ví dụ về một file `/etc/fstab`:

```
/dev/sda1 / ext2 defaults 1 1
/dev/sda2 /usr ext2 defaults 1 1
/dev/sda3 /usr/data ext2 defaults 1 1
/dev/cdrom /cdrom iso9660 ro 1 1
/dev/sda4 /dos msdos defaults 1 1
/dev/sdb1 /data ext2 defaults 1 1
```

Trong đó ta thấy hệ thống file gốc là `/dev/sda1` là hệ thống file Linux điển hình ext2. CD-ROM được nối vào `/cdrom`, nó là hệ thống file theo kiểu *ISO 9660* (CD-ROM) và được nối theo kiểu chỉ đọc.

Linux gắn kết các hệ thống file theo thứ tự của chúng trong file `/etc/fstab`. Ta có thể thấy trong ví dụ trên dòng nối `/usr/data` nằm sau dòng nối `/usr`. Nếu dòng nối

`/usr/data` nằm trước thì lệnh mount tương ứng với nó không thể làm việc vì thư mục `/usr` chưa tồn tại. Trong trường hợp một kết nối không thực hiện được, Linux bỏ qua nó và thực hiện các kết nối còn lại. Nếu một kết nối của một thư mục sẽ được sử dụng trong các kết nối sau không thực hiện được, các kết nối phụ thuộc vào nó cũng sẽ không thực hiện được. Ví dụ nếu kết nối `/usr` bị hỏng vì lý do nào đó, thì kết nối `/usr/data` cũng sẽ không thực hiện được vì thư mục `/usr` không tồn tại.

Hai con số cuối cùng của mỗi dòng trong `/etc/fstab` là tần số sao lưu (dump frequency) và thứ tự kiểm tra (pass number), nói chung không có ý nghĩa với một số phiên bản Linux. Tần số sao lưu chỉ mức độ thường xuyên mà Linux sao lưu lại hệ thống file đó. Nếu là 1 có nghĩa là sao lưu hàng ngày, 2 có nghĩa là sao lưu hai ngày một lần. Con số này được dùng cho các chương trình sao lưu tự động, chương trình này sẽ phân tích file `/etc/fstab` để lấy thông tin này.

Thứ tự kiểm tra là thứ tự mà chương trình `fsck` kiểm tra hệ thống file. Hệ thống file có thứ tự kiểm tra 1 sẽ được kiểm tra đầu tiên, hệ thống file có thứ tự 2 được kiểm tra thứ 2,... Nếu có nhiều hệ thống file có thứ tự kiểm tra bằng 1, các hệ thống file này sẽ được kiểm tra theo thứ tự xuất hiện trong `/etc/fstab`. Hệ thống file gốc phải có giá trị 1, theo lệ thường, các phân hoạch khác được đặt số cao hơn. Tuy vậy, đa số hệ Linux không dùng con số này.

Ta có thể để cả các phân hoạch swap trong `/etc/fstab`. Khi đó đặt thư mục nối đến là none và tần số sao lưu và thứ tự kiểm tra bằng 0. Ví dụ :

```
# /dev/sda2 none swap sw 0 0
```

Khi để phân hoạch swap trong `/etc/fstab`, ta có thể kích hoạt nó bằng lệnh `swapon`

```
# swapon -a
```

Linux đọc file `/etc/fstab` và kích hoạt tất cả các phân hoạch swap. Lệnh này thường được để trong file `/etc/rc` vì thế nó được thực hiện tự động khi khởi động Linux.

Tiếp theo chúng ta sẽ xem xét các kiểu hệ thống file và các giá trị cho trường *option* trong file `/etc/fstab` (tức là hướng dẫn xử lý hệ thống file).

4.2.3 Các kiểu hệ thống file

Các kiểu hệ thống file mà Linux hỗ trợ khác nhau tùy theo phiên bản Linux. Hầu hết các phiên bản hỗ trợ các hệ thống file sau:

ext2	Kiểu hệ thống file mở rộng thứ hai (second extended), là kiểu phổ biến nhất của phân hoạch Linux.
ext	Kiểu hệ thống file mở rộng gốc của Linux(extended), đã bị thay thế bởi ext2.
minix	Kiểu hệ thống file Minix gốc, ít được dùng nhưng vẫn được hỗ trợ vì là khuôn dạng hệ thống file đầu tiên của Linux
xia	Kiểu hệ thống file Xia, ít sử dụng vì bị thay thế bởi ext2.
umsdos	Kiểu hệ thống file UMS-DOS, dùng khi cài đặt Linux trên một phân hoạch DOS (không có phân hoạch riêng cho Linux) .
proc	Kiểu file dựa trên /proc, dùng cho một số tiến trình sử dụng thông tin hệ thống
iso9660	Kiểu hệ thống file ISO 9660 sử dụng trên hầu hết CD-ROM.
xenix	Kiểu hệ thống file SCO Xenix cung cấp hỗ trợ Xenix dưới Linux.
sysv	Kiểu hệ thống file UNIX System V, cho phép hỗ trợ cho ổ System V dưới Linux.
coherent	Kiểu phiên bản Coherent Unix của Mark Williams dùng cho hầu hết các CD-ROM, cho phép hỗ trợ các hệ thống file Coherent dưới Linux
msdos	Kiểu phân hoạch DOS, Linux có thể truy nhập
hpfs	Kiểu hệ thống file High Performance, cho phép hỗ trợ cho HPFS dưới Linux.

Một số phiên bản của Linux không hỗ trợ cho tất cả các hệ thống file trên, đặc biệt là các hệ thống file như *coheren* và *minix*. Hệ thống file *nfs* (Network File System), hỗ trợ hệ thống file mạng, được các phiên bản gần đây của Linux hỗ trợ sẽ được xem xét trong phần sau.

4.2.4 Các giá trị của trường option (hướng dẫn xử lý file)

Trường *option* trong */etc/fstab* có thể có một số giá trị khác nhau, tùy vào phiên bản của Linux. Với hầu hết các phiên bản của Linux (mà dựa trên BSD, UNIX), ta có thể sử dụng các lựa chọn sau để mô tả đặc điểm hệ thống file.

default	Khác nhau tùy theo phiên bản, nhưng thông thường là đọc-ghi, suid và quota
rw	Đọc-ghi
ro	Chỉ đọc

suid	Cho phép truy nhập theo chế độ SUID
nosuid	Không được truy nhập theo chế độ SUID
quota	Quotas may be in effect
noquota	Quotas may not be in effect

Nếu kiểu hệ thống file là nfs, còn có nhiều lựa chọn khác được hỗ trợ. Lựa chọn *default* có xu hướng là lựa chọn tốt nhất cho các hệ thống file truyền thống được nối vào một ổ cứng địa phương.

Thuật ngữ SUID thường dùng khi quản trị hệ thống. SUID là viết tắt của Set User ID và là bit quyền gắn với tất cả các file và thư mục. Cũng có cả bit SGID là viết tắt của Set Group ID. Mọi file hay thư mục có các bit này được thiết lập sẽ hoạt động với quyền của chủ file. Ví dụ, ta đăng nhập hệ thống dưới một tên người dùng thông thường và thực hiện một file nhị phân có bit SUID được thiết lập, file nhị phân sẽ thực thi như nó được chạy bởi root. Cả hai SUID và SGID đều nguy hiểm vì chúng chúng có thể dẫn đến vấn đề an toàn và bảo mật.

4.2.5 Quản lý không gian đĩa

Hệ điều hành, các ứng dụng, chương trình dịch và đặc biệt các file của người dùng trong hệ thống chia sẻ làm cho bao nhiêu không gian đĩa cũng nhanh chóng trở nên không đủ. Đĩa cứng hiện nay không đắt nữa nên nhiều quản trị viên hệ thống chọn giải pháp mua thêm ổ đĩa cứng, điều đó tránh khỏi phải thường xuyên dọn dẹp các file để lấy không gian nhưng vẫn cần có chính sách sử dụng đĩa cho người dùng sao cho không gian đĩa không bị sử dụng một cách lãng phí. Để có được một chính sách sử dụng đĩa như vậy, quản trị viên hệ thống cần phải biết cách xác định không gian đĩa được sử dụng, quản lý đĩa một cách hiệu quả và dọn dẹp đĩa.

4.2.5.1 Kiểm tra hệ thống file

Một phần của trình khởi động Linux (điều khiển bởi một dòng trong */etc/rc*) là kiểm tra tất cả các hệ thống file để đảm bảo chúng không bị hỏng. Công việc này thực hiện mỗi khi khởi động lại hệ thống. Tuy vậy, nếu máy không thường xuyên khởi động lại hoặc bạn là người có kinh nghiệm về về các lỗi của đĩa thì cũng nên tự chạy chương trình kiểm tra hệ thống file.

Nhìn chung, người sử dụng dùng chương trình *fsck* (filesystem check) để kiểm tra hệ thống file. Linux sử dụng một số phiên bản khác của *fsck* để kiểm tra các hệ thống file phụ thuộc Linux, vì vậy, ta có thể không trực tiếp truy nhập *fsck*. Ví dụ nhiều phiên bản Linux có phiên bản riêng của *fsck* để kiểm tra các hệ thống ext2 là *e2fsck*.

Nếu *fsck* có trong một phiên bản của Linux, phiên bản thích hợp của *fsck* được tìm kiếm trong các thư mục */bin*, */sbin*, */etc/fs*, */etc* và thực hiện phiên bản đó.

Quá trình tìm kiếm và thực hiện là trong suốt đối với người sử dụng trong hầu hết các trường hợp.

Fsck thực hiện một số công việc, phần lớn hoạt động của nó là quét toàn bộ hệ thống file để tìm các lỗi sau :

- Một block được dùng bởi nhiều file (vấn đề liên kết chéo : cross-linked) .
- Các block được dùng nhưng lại được đánh dấu là không.
- Các điểm nhập mâu thuẫn giữa các file và bảng i-node.
- Đếm số liên kết không đúng.
- Các điểm nhập không hợp lệ trong bảng i-node.
- Mâu thuẫn giữa giá trị kích thước bảng i-node và không gian được sử dụng bởi các file.
- Các giá trị không hợp lệ trong file.
- Các file đã mất mà không xuất hiện trong bảng i-node.

Toàn bộ quá trình diễn ra rất nhanh, do đó không có lí do gì mà không chạy fsck thường xuyên. Nếu fsck báo cáo các lỗi, hãy đóng hệ thống và chuyển sang chế độ chỉ dành cho siêu người dùng và chạy lại fsck. Các vấn đề có thể đã nảy sinh do ứng dụng cử người dùng, vì vậy bước này sẽ xác định các lỗi kiểu đó. Nếu đĩa vẫn có lỗi, ta có thể sửa nó trong chế độ siêu người dùng.

Đa số các trường hợp, fsck chỉ chạy trên các hệ thống file đã được bỏ nối (unmounted) trừ hệ thống file gốc. Nếu muốn kiểm tra các hệ thống file, trước hết bỏ nối cho nó và sau đó chạy fsck. Để kiểm tra hệ thống file gốc, chuyển hệ thống sang chế độ một người dùng và chạy fsck.

Lệnh fsck lấy tham số là thiết bị hoặc điểm nối hệ thống file mà ta muốn kiểm tra. Ví dụ, cả hai dòng lệnh này đều đúng :

```
fsck /dev/sda1
fsck /usr
```

Nếu fsck làm việc trên nhiều ổ đĩa (nhờ nối), nó sẽ cố gắng thực hiện song song chừng nào có thể để giảm thời gian cần thiết để kiểm tra đĩa.

Các lựa chọn của fsck được hỗ trợ bởi hầu hết các hệ Linux và thường được sử dụng bởi quản trị viên hệ thống là :

-a	Tự động sử dụng hệ thống file mà không cần hỏi lại người dùng
-r	Sửa hệ thống file có tương tác với người sử dụng (hỏi người dùng các chỉ dẫn). Dùng nó chỉ khi kiểm tra một hệ thống file đơn lẻ.

-t <type>	Chỉ ra kiểu hệ thống file kiểm tra. Nếu type bắt đầu bằng no, chỉ các kiểu hệ thống file khác mới được kiểm tra. Lựa chọn này dùng các kiểu hệ thống file trong file /etc/fstab.
-v	Lựa chọn này cung cấp đầu ra dài (báo cáo dài)

Nên quen với việc thỉnh thoảng chạy fsck chỉ để kiểm tra tính chặt chẽ của hệ thống. Mỗi khi gặp một thông báo lỗi đĩa, hãy chạy fsck ngay.

4.2.5.2 *Hiển thị các thống kê về hệ thống file*

Hai lệnh thường được dùng để xem các thống kê hệ thống file nhất là *df* (disk filesystem) và *du* (disk usage), chúng cho biết không gian đã dùng, không gian còn trống, ...

Lệnh df

Lệnh *df* được dùng nhiều nhất để lấy các thống kê về hệ thống file. Nó hiển thị các thông tin về tất cả các hệ thống file trong hệ thống, dung lượng tổng cộng, lượng còn trống trên mỗi hệ thống file, các vị trí nổi hiện tại. Ví dụ, một lệnh *df* cho kết quả :

```
$ df
Filesystem 1024-blocks Used Available Capacity Mounted on
/dev/sda3 478792 94598 359465 21% /
/dev/sda1 511712 44288 467424 9% /dos
/dev/scd0 663516 663516 0 100% /cdrom
```

Như vậy hệ thống này có một đĩa cứng SCSI với 2 phân hoạch, một cho Linux và một cho DOS, và một CD-ROM. Phân hoạch Linux /dev/sda3, có dung lượng tổng cộng 478.792K, đã dùng 94.598K, không gian còn trống 359.465K, phân hoạch này đã sử dụng 21%, và nó là phân hoạch gốc. Tương tự với các phân hoạch sau.

Thỉnh thoảng chúng ta thấy thông báo phần trăm sử dụng đĩa là 100%. Đó là do Linux giữ lại 10% đĩa cho siêu người dùng sử dụng, như vậy, root có thể sử dụng 110% dung lượng được hiển thị. Tuy vậy khi lượng sử dụng lên đến 100% thì cũng là lúc phải dọn dẹp đĩa.

Lựa chọn *-i* của lệnh đưa ra các thông tin tương tự về bảng i-node

```
$ df -i
Filesystem Inodes IUsed IFree %IUsed Mounted on
```

/dev/sda3	123952	8224	115728	7%	/
/dev/sda1	0	0	0	0%	/dos
/dev/scd0	0	0	0	0%	/cdrom

Cũng với hệ thống như trên, df đưa ra các con số về các i-node sẵn có, lượng đã dùng, số còn rồi, và phần trăm đã sử dụng. Không có sự tương quan giữa thông tin sử dụng bảng i-node và thông tin sử dụng đĩa, vì vậy nên hiển thị cả hai thông tin. Một i-node được sử dụng khi một file được dùng, vì vậy nếu nhiều file nhỏ được lưu thì bảng i-node cũng có thể đầy. Do đó phải kiểm tra cả tình trạng sử dụng đĩa và bảng i-node.

Lệnh df bỏ qua các hệ thống file được đánh dấu là bỏ qua trong file /etc/fstab (thường chỉ có các file swap có thiết lập này). Mặc định, lệnh df hiển thị tất cả các hệ thống file được nối vào hệ thống, trừ phi ta đặc tả một hệ thống file cho lệnh.

Lệnh df hiển thị không gian sử dụng đĩa theo các block 1KB, trừ phi ta đặt biến môi trường POSIXLY_CORRECT trong file khởi tạo hệ thống, nếu biến này được thiết lập, thông tin sẽ được hiển thị theo block 512 byte. Cách đặt này thuận lợi khi dùng các hệ thống file cũ sử dụng các sector với kích thước 512 byte.

Sau đây là một số lựa chọn của lệnh df, hầu hết được các phiên bản của Linux hỗ trợ :

-a, -all	hiển thị cho cả các hệ thống file không có block nào (Các hệ thống file không có block nào thường dùng cho các mục đích đặc biệt như tự động nối các thiết bị đặc biệt.)
-help	Hiển thị các thông tin trợ giúp
-i, -inode	Hiển thị các thông tin i-node
-k, -kilobyte	Dùng để hiện không gian theo block 1K thay vì block 512 byte
-p 4	Dùng khuôn dạng POSIX để hiển thị thông tin về hệ thống file trên một dòng mà không có cuộn dòng. Nếu tên hệ thống file dài quá 20 ký tự, lựa chọn này sẽ bắt buộc các cột không căn lề nữa.
-T	Hiển thị thêm thông tin kiểu của các hệ thống file.
-t<type>	Chỉ hiển thị các hệ thống file có kiểu phù hợp với kiểu đưa ra.
-v	Hiển thị số phiên bản của df
-x<type>	Hiển thị mọi hệ thống file khác kiểu đã chỉ ra.

Các lựa chọn trên có thể được tổ hợp với nhau khi dùng. Ta cũng có thể đưa các lệnh thường dùng với nhau vào một shell script để chạy mỗi khi cần.

Lệnh du

Lệnh *du* dùng để hiển thị các thông tin thống kê tình trạng sử dụng đĩa có ích. Khi chạy lệnh *du* không có tham số, nó hiển thị lượng đĩa đã dùng bởi các file, thư mục con dưới một thư mục đưa ra hay thư mục hiện tại nếu không đưa ra thư mục nào. Hãy xem hai ví dụ sau:

```
$ du
125 /info/a_temp
4 /info/data
265 /info/data/book
726 /info/data/book/chap_1
```

Và:

```
$ du /usr/tparker
35 /usr/tparker/bin
2736 /usr/tparker/book
3 /usr/tparker/source
```

du hiển thị không gian sử dụng của mỗi thư mục theo block trong cột đầu và tên thư mục trong cột thứ hai.

Nếu chạy *du* trên một cây thư mục lớn, thông tin đầu ra sẽ rất dài, ta có thể tóm tắt nó bằng lựa chọn *-s* (summarize):

```
$ du -s /usr/ychow
3315 /usr/ychow
```

du có một số lựa chọn sau :

-a, -all	Hiển thị thông tin cho mọi thư mục và file
-b, -bytes	Hiển thị kích thước theo byte
-c, -total	Hiển thị toàn bộ
-k	Hiển thị kích thước theo kilobyte , thay vì 512 byte (nếu trước đó đã

	đặt là 512 byte)
-l	Hiển thị kích thước của tất cả các file, kể cả các file link
-s	Chỉ hiển thị tổng số
-v	Hiển thị phiên bản
-x	Bỏ qua các thư mục trong các hệ thống file khác được nối vào hệ thống file hiện tại.

4.2.5.3 Dọn dẹp đĩa cứng

Khi thiếu không gian đĩa, tất nhiên ta có thể dùng giải pháp mua thêm ổ đĩa thêm phân hoạch cho Linux, nhưng như vậy không được thực tế lắm và cũng không phải điều ta mong muốn. Giải pháp trong trường hợp này là xoay xở với những gì mà chúng ta có.

Hệ thống hoạt động kém trước tiên là vì bị phân mảnh đĩa, các đầu đọc phải di chuyển nhiều để truy nhập hay ghi một file. Vì vậy cần loại bỏ sự phân mảnh đĩa.

Để giảm không gian sử dụng đĩa, đầu tiên nên loại bỏ các ứng dụng và phần mềm không cần dùng đến, nhất là những phần mềm được cài đặt tự động khi cài đặt Linux, chẳng hạn như khi gỡ bỏ bộ dịch C, không gian đĩa có thể giảm được tới 50MB.

Tiếp theo, yêu cầu người dùng hệ thống dọn dẹp không gian dành cho họ, xoá các bản sao của các file, bỏ các file lưu tự động và các file log. Chỉ riêng việc xoá bỏ các file log cũng có thể tiết kiệm không gian đĩa tới 30MB. Các file log cơ bản bao gồm:

/usr/spool/lp/log	printing log
/usr/lib/cron.log	cron log file
/usr/spool/uucp/LOGFILE	UUCP log file

Kích thước các file này có thể tăng nhanh một cách đáng kinh ngạc. Nếu muốn giữ lại một số dòng trong file log, hãy dùng lệnh tail. Ví dụ để giữ lại 100 dòng đầu còn xoá các dòng còn lại.

```
cd /usr/spool/lp
tail -100 log > tmp
mv tmp log
```

Sao lưu các chương trình, dữ liệu không hay dùng đến vào các thiết bị nhớ khác và xoá chúng khỏi ổ cứng. Nếu thật cần thiết phải giữ chúng trong ổ cứng, hãy nén lại.

Nên nhớ rằng file *core*, *log*, *.error* là các file mà kích thước tăng nhanh đến khá lớn và chúng ta có thể xóa đi mà chẳng hại gì. Để xóa đi các file này, ví dụ tất cả các file *core*, ta dùng lệnh

```
find / -name core -exec rm {} \;
```

4.2.6 File liên kết (link)

File liên kết là một khái niệm thường gây nhầm lẫn trong hệ thống file nhưng thực ra nó rất đơn giản. Một cách đơn giản nhất, một liên kết tạo một tên file thứ hai cho một file. Có hai loại file liên kết, liên kết cứng (hard link) và liên kết biểu tượng (symbolique link).

Một liên kết cứng là một liên kết trong cùng hệ thống file với hai điểm i-node cùng trỏ đến một nội dung vật lý (cùng số i-node vì cùng trỏ đến một dữ liệu).

Ví dụ ta tạo một file liên kết của file */usr/bill/testfile* bằng lệnh sau :

```
ln /usr/bill/testfile /usr/tim/testfile
```

Khuôn dạng của lệnh này luôn là tên file hiện tại rồi đến tên file liên kết.

Khi đó để biết số i-node của 2 file, ta dùng

```
$ ls -li /usr/bill/testfile /usr/tim/testfile
14253 /usr/bill/testfile 14253 /usr/tim/testfile
```

Cả hai file */usr/bill/testfile*, */usr/tim/testfile* cùng trỏ đến một nội dung vật lý chung vì vậy mọi thay đổi bởi bill hoặc tim đều phản ánh ngay lập tức trên thư mục kia (việc này tránh cho chúng ta phải thường xuyên sao lại các file này). Cả bill và tim đều có thể thay đổi file, miễn là không thay đổi cùng một lúc.

Một vấn đề là quyền truy nhập file và sở hữu file như thế nào. Nếu bill là người chủ file và là người duy nhất có quyền ghi file, anh ta có thể tạo một móc nối đến */usr/tim/testfile* và đặt quyền sở hữu của file liên kết mới cho tim. Theo cách đó cả bill và tim có thể làm việc trên cùng một file bất chấp các quyền sở hữu và truy nhập vì mỗi file có một người sở hữu riêng. Nếu được đặt đúng, quyền sở hữu và quyền truy nhập có thể chống những người khác đọc và ghi lên file.

Việc xóa một tên file liên kết không xóa mất file vật lý tương ứng, nó chỉ bị xóa khi nào không còn một liên kết nào đến nó nữa.

Liên kết biểu tượng là một kiểu khác của liên kết mà không dùng điểm i-node cho liên kết. Ta đã sử dụng các liên kết này khi tạo các trình điều khiển thiết bị ngoại vi như */dev/modem* hay */dev/cua1*. Lựa chọn *-s* của lệnh *ln* cho phép tạo liên kết biểu tượng. Ví dụ có file *bigfile*, ta tạo một liên kết biểu tượng của nó, file *anotherfile*.

```
$ ls -li bigfile
```

```
6253 bigfile
$ ln -s bigfile anotherfile
$ ls -li bigfile anotherfile
6253 bigfile 8358 anotherfile
```

Như ta thấy các số i-node của các file là khác nhau. Khi liệt kê file, các liên kết biểu tượng được biểu diễn bằng mũi tên:

```
lrwxrwxrwx 1 root root 6 Sep 16:35 anotherfile->bigfile
-rw-rw-r-- 1 root root 2 Sep 17:23 bigfile
```

Quyền của file liên kết biểu tượng luôn được đặt là lrwxrwxrwx. Quyền truy nhập file liên kết biểu tượng được xác định bởi quyền truy nhập và sở hữu của file mà nó liên kết đến (ở đây là bigfile).

Điểm khác biệt giữa liên kết cứng và liên kết biểu tượng không chỉ là các điểm nhập i-node trong bảng i-node. Ta có thể tạo một liên kết biểu tượng đến một file chưa tồn tại, điều này không thể làm được với liên kết cứng. Ta có thể đi theo các liên kết biểu tượng để tìm ra file mà chúng trỏ tới nhưng hầu như không thể làm được với liên kết cứng. Hạt nhân Linux xử lý hai kiểu file này cũng khác nhau.

4.3 QUẢN LÝ NGƯỜI DÙNG

Mọi truy nhập vào hệ thống Linux đều thông qua một tài khoản của người sử dụng (user account). Mỗi tài khoản được thiết lập bởi quản trị viên hệ thống ngoại trừ tài khoản root (và một số tài khoản hệ thống mà người sử dụng ít khi dùng tới). Mặc dù một số hệ Linux chỉ có một người dùng nhưng cũng không nên dùng tài khoản root cho các truy nhập hàng ngày. Hầu hết các hệ thống đều cho phép nhiều người truy nhập hoặc qua đa người dùng trong console chính, qua modem, mạng, hoặc các thiết bị đầu cuối được nối cứng. Vì vậy phương pháp đặt và quản lý các tài khoản, các thư mục cũng như các file liên quan đến nó là một khía cạnh quan trọng của quản trị hệ thống Linux.

4.3.1 Tài khoản siêu người dùng (superuser account)

Khi cài đặt phần mềm Linux, một tài khoản chủ được tạo tự động, đó là root, (cũng được biết dưới tên siêu người dùng vì khi đăng nhập vào tài khoản này người sử dụng có quyền sử dụng tuyệt đối trên hệ thống). Các tài khoản người sử dụng của Linux đều bị hạn chế một số quyền để tránh người dùng chẳng may làm hỏng hệ thống

Khả năng của tài khoản root là không giới hạn, khi đăng nhập với account root người sử dụng có thể thêm các quyền mới cho nó. Chính vì root có sức mạnh như vậy cho nên nó rất hấp dẫn đối với người mới sử dụng, và khi sử dụng hệ thống với account này họ rất dễ gây rối loạn hệ thống. Do vậy chỉ nên sử dụng account root để bảo trì hệ thống chứ không nên sử dụng nó cho các mục đích hàng ngày. Hãy suy nghĩ kĩ khi sử dụng mỗi lệnh của root và nên nhớ rằng trong một hệ thống đa người dùng thì việc làm hỏng hệ thống sẽ làm ảnh hưởng đến nhiều người khác.

Vì vậy, tốt hơn cả là đối với các hệ thống mới, mỗi người sử dụng cần có một tài khoản riêng để truy nhập hàng ngày, còn với account root hãy đặt cho nó một mật khẩu mà những người sử dụng khác không đoán được.

Chúng ta cũng có thể tạo một số tài khoản đặc biệt mà không cần nhiều quyền truy nhập cho quản trị hệ thống, chẳng hạn để sao lưu ra băng từ. Ta có thể thiết lập một tài khoản chỉ có quyền đọc của root tới toàn bộ hệ thống file để tránh gây hỏng hóc. Tài khoản này có thể dùng để sao lưu hệ thống đồng thời tránh chúng ta chẳng may xóa mất hạt nhân. Có thể thiết lập một số tài khoản tương tự cho truy nhập e-mail, gateway vào Internet,...

Một điều quan trọng là không phải tài khoản siêu người dùng nào cũng gọi là root, mặc dù nó được tạo mặc định là root khi cài đặt Linux, nó có thể có tên bất kì nhưng thường được dùng nhất dưới tên root. Tài khoản này được định nghĩa là tài khoản có userID là 0, các userID được định nghĩa trong `/etc/passwd`.

4.3.2 Thiết lập tài khoản người sử dụng

Để hệ thống có thể được nhiều người sử dụng ta phải tạo tài khoản cho mỗi người trong số họ. Ta cũng có thể muốn tạo một tài khoản khách (guest) cho nhiều người sử dụng mà không cần mật khẩu.

Mỗi người dùng Linux phải có một tên sử dụng, một mật mã riêng, trừ tài khoản guest hoặc tài khoản truy nhập một ứng dụng đặc biệt như cơ sở dữ liệu. Bằng cách cấp cho mỗi người sử dụng một tài khoản riêng, mức độ bảo mật của hệ thống sẽ được nâng lên và quản trị viên hệ thống có thể biết rõ ai truy nhập hệ thống và đang làm gì. Ánh xạ một người dùng và một tài khoản cho phép theo dõi các hoạt động dễ hơn nhiều.

File `/etc/passwd` chứa các thông tin về tài khoản của người sử dụng, nó nên được đặt dưới quyền sở hữu của root và groupID đặt bằng 0 (0 thường dùng để chỉ root hoặc nhóm hệ thống, được định nghĩa trong file `/etc/group`). Nên đặt quyền để chỉ root có quyền ghi còn người dùng khác chỉ có quyền đọc.

Các dòng trong file `/etc/passwd` đều có khuôn dạng :

```
username:password:userID:groupID:comment:home
directory:login command
```


Ví dụ một file `/etc/passwd` được tạo khi Linux mới được cài đặt:

```
root::0:0:root:/root:/bin/bash
bin:*:1:1:bin:/bin:
daemon:*:2:2:daemon:/sbin:
adm:*:3:4:adm:/var/adm:
lp:*:4:7:lp:/var/spool/lpd:
sync:*:5:0:sync:/sbin:/bin/sync
shutdown:*:6:0:shutdown:/sbin:/sbin/shutdown
halt:*:7:0:halt:/sbin:/sbin/halt
mail:*:8:12:mail:/var/spool/mail:
news:*:9:13:news:/usr/lib/news:
uucp:*:10:14:uucp:/var/spool/uucppublic:
operator:*:11:0:operator:/root:/bin/bash
games:*:12:100:games:/usr/games:
man:*:13:15:man:/usr/man:
postmaster:*:14:12:postmaster:/var/spool/mail:/bin/
bash
nobody:*:-1:100:nobody:/dev/null:
ftp:*:404:1::/home/ftp:/bin/bash
```

mỗi dòng đều có 7 trường phân cách nhau bởi dấu ':'. Nếu 1 trường không có giá trị thì trường đó bị bỏ trống. Các trường lần lượt có ý nghĩa như sau :

username : tên duy nhất xác định một người sử dụng

password : mật khẩu của người sử dụng đã được mã hoá

user ID (UID): số duy nhất xác định một người đối với hệ điều hành.

GroupID (GID) : số duy nhất xác định nhóm của người dùng được xác định trong dòng này (phục vụ cho đặt quyền trên file).

Comment : Các thông tin ghi chú, thường dùng là tên thật của người sử dụng, nhưng có thể là thông tin khác.

Home directory : thư mục mà người sử dụng sẽ được đặt vào khi họ đăng nhập (login) vào hệ thống.

Login command : Lệnh sẽ thực hiện ngay sau khi người sử dụng đăng nhập vào hệ thống, thông thường là lệnh gọi chương trình shell

Bây giờ ta sẽ tìm hiểu chi tiết về từng trường và các chương trình sử dụng các trường này như thế nào trong Linux. File kiểu này được sử dụng trong hầu hết các hệ Unix, do đó các hiểu biết về nó trên Linux cũng sẽ cho ta các hiểu biết trên hầu hết các hệ Unix khác.

4.3.2.1 Username (tên người dùng)

Tên người dùng có thể là một xâu kí tự, thường là 8 hoặc nhỏ hơn, nó xác định duy nhất một người dùng. Tên này là phương tiện thường dùng nhất để giao tiếp giữa các người dùng với nhau và với các máy khác, vì vậy nên đơn giản, dễ đoán. Người ta có thể dùng bất kì cách đặt tên nào cho username của mình, miễn là đảm bảo tên đó là duy nhất trong mạng, nhưng đôi khi các tên độc đáo lại làm cho những người dùng khác, máy khác không biết họ là ai và gây khó khăn cho giao tiếp, hoặc các tên quá đơn giản như kiểu chỉ gồm tên thật không có họ, lại dễ bị trùng trong các hệ thống mạng lớn. Thông thường tên này là một hoán vị của tên thật của người dùng, thường là tổ hợp của tên viết tắt và họ đối với người nước ngoài và họ viết tắt cộng với tên đối với người Việt Nam. Ví dụ Tomat Parker sẽ có username : tparker, hay Nguyễn Văn Việt có username là nvviet.

Linux cũng như mọi hệ Unix khác có phân biệt chữ hoa và chữ thường, vì vậy các tên username viết hoa và viết thường sẽ được hiểu là hai tên khác nhau. Vì hầu hết các lệnh của Linux đều được viết bằng chữ thường nên chúng ta nên để tên người dùng ở dạng chữ thường. Các kí tự gạch dưới, cách, số, và một vài kí tự đặc biệt có thể được dùng trong tên người dùng nhưng lại làm cho tên trông có vẻ lạ và đôi khi gây rắc rối cho một vài ứng dụng.

4.3.2.2 Password (mật khẩu)

Hệ thống lưu mật khẩu đã được mã hoá của người dùng vào trường password. Chỉ quản trị viên hệ thống mới có thể đổi mật khẩu, hoặc chính người sử dụng tự đổi mật khẩu của mình bằng lệnh passwd, mọi thay đổi trực tiếp khác trên trường này đều có thể làm cho tài khoản tương ứng không thể sử dụng được nữa.

Một số phiên bản Unix không để mật khẩu đã mã hoá trong file /etc/passwd vì vấn đề bảo mật, trong trường password ta chỉ thấy thấy chữ x, mà để trong file /etc/shadows. Nhưng nói chung thì trường này mặc định vẫn được để .

Khi người sử dụng đăng nhập vào hệ thống, chương trình login mã hoá mật khẩu được nhập, đem so sánh kết quả với trường password của người đó. Đăng nhập sẽ không thực hiện được nếu có sự khác nhau.

Nếu kí tự đầu tiên của trường password là '*' thì có nghĩa là tài khoản này không được sử dụng để truy nhập hệ thống, dấu * hạn chế mọi truy nhập. Như thấy trong file /etc/passwd ở trên, có rất nhiều tài khoản có trường password là dấu *.

Nếu trường password bị bỏ trống thì mọi người sử dụng đều có thể đăng nhập tài khoản này một cách không hạn chế mà không yêu cầu mật khẩu.

4.3.2.3 User ID (định danh người sử dụng)

Mọi người sử dụng đều được tương ứng với một số hiệu duy nhất gọi là UID, dùng để xác định mọi thứ có liên quan đến người sử dụng đó. Chẳng hạn Linux sử dụng định danh này để theo dõi mọi tiến trình do một người sử dụng kích hoạt. Định danh người sử dụng thường được sử dụng nhiều hơn tên vì chúng chỉ gồm những con số, dễ làm việc hơn là chuỗi kí tự tên người sử dụng, và cũng tốn ít không gian lưu trữ hơn. Có một số tiện ích cho phép chuyển đổi từ định danh người sử dụng sang tên bằng cách duyệt file /etc/passwd và tìm đến UID phù hợp để lấy tên.

Hầu hết các hệ Unix dùng các số từ 0 đến 99 cho các tài khoản dành cho hệ thống, và từ 100 trở đi dành cho người sử dụng. Như ta thấy trong file ví dụ ở trên root có UID bằng 0, các tài khoản khác của hệ thống dùng các số lớn hơn. Đặc biệt, nobody là một tài khoản đặc biệt chỉ dùng cho NFS (Network File System) có UID -1. Khi gán số hiệu cho người sử dụng, chúng ta gán lần lượt từ 100, 101,...

4.3.2.4 Group ID (số hiệu nhóm)

Số hiệu nhóm được dùng để xác định nhóm mà người sử dụng thuộc vào khi đăng nhập vào hệ thống. Nó dùng cho mục đích đặt quyền trên file. Số hiệu nhóm là các số bắt đầu từ 0, đa số các hệ thống Unix dùng các số từ 0 đến 49 (hoặc 0 đến 9) cho các nhóm hệ thống và từ 50 trở đi cho người dùng. Nhóm mặc định là 50.

Các nhóm và số hiệu nhóm được liệt kê trong file /etc/group.

4.3.2.5 Comment (thông tin ghi chú)

Các quản trị viên hệ thống dùng trường này để cung cấp thêm thông tin cần thiết cho mỗi tài khoản tương ứng trong /etc/passwd, làm cho chúng trở nên dễ hiểu hơn. Nó thường chứa tên đầy đủ của người sử dụng, nhưng đôi khi là phòng ban hoặc một con số mở rộng nào đó tùy theo quản trị viên. (Trường này đôi khi được gọi là GECOS, General Electric Comprehensive Operating System, tùy thuộc vào hệ điều hành dùng nó).

Một số tiện ích dùng trường này để hiển thị thông tin về người sử dụng, do vậy không nên để các thông tin nhạy cảm với việc sửa đổi ở đây. Chẳng hạn hệ thống e-mail truy nhập vào nó để đưa ra ai là người gửi thư đi.

4.3.2.6 Home Directory (Thư mục cá nhân)

Trường này chỉ cho tiến trình login nơi đặt người sử dụng khi họ đăng nhập vào hệ thống (tức là thư mục làm việc khởi đầu của người đó), thường là thư mục cá nhân của người đó. Mỗi người dùng nên có một thư mục cá nhân và file khởi động của họ sẽ khởi tạo cho biến môi trường HOME giá trị thư mục này. Người sử dụng không bị hạn chế gì trong thư mục chỉ ra trong trường home directory ngoại trừ một số quyền trên file được đặt để tránh di chuyển.

Nhìn chung thư mục cá nhân của người sử dụng được đặt trong một vùng chung, Unix có xu hướng dùng thư mục `/home`, do đó người sử dụng có tài khoản nvviet sẽ có thư mục cá nhân `/home/nvviet`. Một số phiên bản khác của Unix dùng thư mục `/usr`, `/user` hoặc `/u` để chứa các thư mục cá nhân, hoặc một số nhà quản trị hệ thống lại thích dùng cấu trúc thư mục khác để thuận tiện cho họ. Linux không quan tâm lắm đến vấn đề này, chỉ cần thư mục đó có thể vào được.

4.3.2.7 *Lệnh Login*

Trường này chỉ lệnh được thực hiện ngay khi chương trình login kết thúc. Lệnh này thường là một lệnh shell kích hoạt một chương trình C shell hoặc Bourne shell nhằm cung cấp cho người dùng một môi trường shell. Trong một số trường hợp nó có thể là một ứng dụng đơn giản hoặc một ứng dụng đầy đủ của hệ thống để hạn chế những gì người dùng có thể thực hiện. Ví dụ tài khoản uucp (sử dụng cho e-mail hoặc cho các công việc về mạng đơn giản khác) chỉ thực hiện lệnh uucp. Nếu trường này rỗng, hệ điều hành mặc định lệnh là Bourne shell (mặc định này cũng tùy theo hệ điều hành được cài đặt).

Nhiều phiên bản của Linux cho phép người dùng thay đổi shell khởi đầu (sau khi vào hệ thống) bằng lệnh `chsh` hoặc `passwd -s`. Khi nhận lệnh này Linux tìm trong file `/etc/shells` đến một shell phù hợp. Chỉ các lệnh trong `/etc/shells` mới là những lệnh mà người dùng được phép dùng thay thế cho lệnh gọi shell khởi đầu được đặc tả trong trường login command. Điều này làm cho hệ thống chặt chẽ hơn về mặt an toàn. Các dòng trong `/etc/shell` có thể thêm hoặc bớt đổi bằng các trình soạn thảo. Chúng ta nên để cho file này có cùng quyền truy nhập và cùng người sở hữu như `/etc/passwd` để tránh người dùng sửa lệnh khởi đầu khi họ đăng nhập vào hệ thống.

4.3.2.8 *Một số tên người dùng mặc định của hệ thống*

Như thấy trong file `/etc/passwd`, có rất nhiều tên người dùng của hệ thống. Một số trong đó được dùng cho hệ điều hành và quản trị viên.

Tài khoản *root* : tài khoản siêu người dùng (UID 0) không bị hạn chế quyền truy nhập, nó là chủ của nhiều file hệ thống.

Tài khoản *daemon* : được sử dụng cho các tiến trình hệ thống, chỉ được dùng để làm chủ các tiến trình và đặt các quyền của nó một cách chính xác.

Tài khoản *bin* : Làm chủ các file thực thi được

Tài khoản *sys* : làm chủ các file thực thi được

Tài khoản *adm* : làm chủ các file sổ sách (các file ghi lại thông tin hoạt động, các quá trình của hệ thống) và file log

Tài khoản *uucp* : dùng cho các giao tiếp và file UUCP

Ngoài ra còn có một số tài khoản khác dùng cho mục đích riêng như *postmaster* cho e-mail... Ta không nên thay đổi bất kì tài khoản nào của hệ thống, nói chung chúng đều có trường password là dấu * để tránh truy nhập.

4.3.2.9 Thêm người dùng

Ta có thể thêm người sử dụng hệ thống bằng cách sửa file */etc/passwd* bằng tay hoặc sử dụng một script tự động nhắc chúng ta các thông tin chi tiết về người dùng mới phải nhập và ghi dòng thông tin mới vào file */etc/passwd* cho chúng ta. Script này làm cho công việc thêm người dùng trở nên dễ dàng hơn và tránh được các lỗi gặp phải khi phải thêm nhiều người dùng bằng tay. Như chúng ta đã biết, ta chỉ có thể thay đổi */etc/passwd* khi ta có quyền root vì vậy phải đăng nhập dưới tên root. Để cho an toàn, hãy sao lưu */etc/passwd* trước khi sửa đổi nó, nếu không khi vô tình làm hỏng file này chúng ta không thể vào lại hệ thống kể cả từ tài khoản root, và hệ thống chỉ có thể làm việc trong chế độ quản trị hệ thống.

Để thực hiện bằng tay, quá trình thêm người sử dụng gồm có :

- Thêm dòng tương ứng với người mới vào file */etc/passwd*
- Tạo thư mục cá nhân cho người này (tất nhiên trùng với thư mục cá nhân đã chỉ ra trong */etc/passwd*)
- Chép các file khởi động shell và sửa đổi các thiết lập và quyền sở hữu các file đó

Đầu tiên, để thêm một dòng của người sử dụng mới vào file */etc/passwd*, ta phải dùng một trình soạn thảo mà nó ghi thông tin dưới dạng mã ASCII. Thêm người mới vào cuối file, mỗi người một dòng. Lưu ý là tên người dùng và số hiệu người dùng phải là duy nhất với mỗi người. Ví dụ thêm một người với tên người dùng là *bill*, theo thứ tự tuần tự người này có UID là 103, thuộc nhóm mặc định *GID=50*, thư mục cá nhân mà ta sẽ tạo là */home/bill*, và shell khởi đầu là Bourne shell (do đó lệnh gọi shell là */bin/sh*)

```
bill::103:50:Bill Smallwood:/home/bill:/bin/sh
```

Chúng ta để trường password trống vì ta không thể tạo ra một mật mã đã mã hoá được. Sau khi ghi file này, ta sẽ tạo cho người này một mật khẩu tạm thời bằng lệnh

```
# passwd bill
```

Lệnh này sẽ nhắc ta nhập mật khẩu cho người này và yêu cầu anh ta phải đổi mật khẩu ngay lần đăng nhập hệ thống đầu tiên. Thông thường mật khẩu tạm thời là một xâu chung chung, có thể là tên đăng nhập hệ thống (tên người dùng).

Mỗi người dùng đều thuộc về một nhóm, ta thêm tên người dùng mới vào file `/etc/group` tại dòng biểu diễn nhóm của họ (nếu người này thuộc nhiều nhóm thì phải thêm vào tất cả các dòng định nghĩa các nhóm đó).

Tiếp theo, ta tạo thư mục cá nhân cho họ và dùng quyền đặt cho họ quyền làm chủ thư mục. Theo như trên thì lệnh phải thực hiện sẽ là :

```
# mkdir /home/bill
# chown bill /home/bill
```

Cuối cùng, chép các file cấu hình cho shell vào thư mục cá nhân của họ và đặt hệ thống để họ có thể sửa đổi file này theo ý mình. Nếu ta chép file này từ một người dùng khác là yvonne chẳng hạn thì các lệnh phải dùng là :

```
# cp /home/yvonne/.profile /home/bill/.profile
# chown bill /home/.profile
```

Kiểm tra lại bằng tay file cấu hình để đảm bảo rằng các biến môi trường không bị đặt sai khi người sử dụng đăng nhập (việc kiểm tra này cũng phải thực hiện bằng một trình soạn thảo ASCII) . Ví dụ dòng định nghĩa biến HOME, hay thư mục spool cho máy in và e-mail. Nếu dùng Bourne shell và các phiên bản tương thích thì chỉ cần file `.profile`, nhưng nếu dùng C shell và các bản tương thích với nó thì cần `.login` và `.cshrc`. Korn và các bản tương thích cần `.profile` và thường một file khác với các biến môi trường nhúng trong đó.

Hệ Linux có một lệnh lấy từ phiên bản Berkeley BSD UNIX, lệnh `vipw` cho phép soạn thảo một bản sao tạm của `/etc/passwd` bằng vi (hoặc một trình soạn thảo mặc định khác). Cách dùng một file tạm và khoá file tạo thành một cơ chế chống nhiều người cùng sửa một file đồng thời. Khi ghi file `vipw` thực hiện một kiểm tra đơn giản trên file thay đổi, nếu tất cả đều đúng file `/etc/passwd` được cập nhật.

Script tự động thực hiện việc thêm người dùng có tên `useradd` hoặc `adduser`. Khi chạy chúng sẽ nhắc chúng ta các thông tin cần thiết cho file `/etc/passwd`. Chúng đều yêu cầu nhập mật mã, chúng ta có thể bỏ trống. Một ưu điểm của script tự động là nó tự chép các file hỗ trợ shell, và trong một số trường hợp nó thay đổi biến môi trường cho chúng ta. Những script này làm đơn giản quá trình thêm một số lượng lớn người dùng mới.

4.3.2.10 Xoá người dùng

Giống như thêm người dùng mới, ta có thể xoá người dùng bằng một script tự động hoặc bằng tay. Script tự động `deluser` hoặc `userdel` sẽ hỏi chúng ta muốn xoá người dùng nào, sau đó xoá điểm nhập (dòng) của người dùng đó trong file `/etc/passwd`. Một số script xoá cả spool và các file trong thư mục cá nhân nếu

muốn. Để thực hiện việc loại bỏ người dùng tất nhiên ta phải đăng nhập dưới tên root để có thể xoá trong file `/etc/passwd`.

Quá trình xoá người dùng bằng tay (hoặc bằng script tự động mà không xoá các thư mục và file) như sau :

- Xoá điểm nhập tương ứng với người dùng trong `/etc/passwd` và trong `/etc/group`
- Xoá các file mail và các mail alias của người dùng.
- Xoá mọi cron và at.
- Xoá thư mục cá nhân nếu không muốn giữ các file trong đó.

Để xoá người dùng bằng tay, đầu tiên phải xoá điểm nhập tương ứng trong `/etc/passwd`. Đơn giản là chúng ta xoá dòng tương ứng trong đó, nếu muốn giữ lại người dùng đó vì một lí do nào đó (chẳng hạn vì vấn đề sở hữu file, vì một tài khoản dùng chung, vì vấn đề lưu trữ tính toán...), ta chỉ cần làm cho tài khoản của họ không truy nhập được bằng thay trường `passwd` của họ trong `/etc/passwd` bằng dấu `*`. Như vậy muốn khôi phục lại ta phải dùng lệnh `passwd` để thay đổi mật mã cho họ.

Tiếp theo phải xoá thư mục cá nhân của người này bằng lệnh sau :

```
# rm -r /home/userdir
```

với `/home/userdir` là đường dẫn đầy đủ của thư mục cá nhân.

Sau đó xoá file `spool` của mail (chỉ có 1 file), file này thường nằm trong thư mục `/usr/spool/mail`. Ví dụ muốn xoá của người `walter`, ta viết lệnh.

```
# rm /usr/spool/mail/walter
```

Để kết thúc việc xoá mail, kiểm tra xem người dùng đó có điểm nhập nào trong file mail alias không (thông thường là file `/usr/lib/aliases`), nếu có ta có thể bắt tất cả các mail cho người dùng đó vào một tài khoản khác (như `root`) với một điểm nhập trong file `aliases`. Cuối cùng chắc chắn rằng không còn điểm nhập nào trong file `cron` và `at` của người dùng này mà hệ thống sẽ tiếp tục thực hiện. Để hiển thị file `crontab`, dùng lệnh `crontab`.

Nếu muốn xoá tạm thời một tài khoản, ta chỉ việc vô hiệu việc đăng nhập vào nó bằng cách thêm dấu `*` vào đầu trường `passwd` tương ứng trong `/etc/passwd` (vẫn giữ nguyên `passwd`), khi muốn khôi phục lại ta chỉ việc xoá dấu `*` này đi.

4.3.3 Sử dụng nhóm

Mọi người dùng trong các hệ UNIX và Linux đều thuộc về một nhóm. Một nhóm là một tập hợp các cá nhân đơn lẻ được gộp lại vì một lý do nào đó tùy ý. Những người dùng trong nhóm có thể làm cùng phòng, có thể cùng cần truy nhập một chương trình tiện ích, hoặc cùng truy nhập đến một thiết bị đặc biệt như máy in, scanner... mỗi người có thể thuộc nhiều nhóm, tuy vậy, họ chỉ có thể là thành viên của một nhóm tại một thời điểm khi nhóm được dùng để xác định quyền trên file và Linux chỉ cho phép có một group ID (số hiệu nhóm) trong một thời điểm.

Các nhóm được đặt quyền để các thành viên của nó có thể truy nhập đến các thiết bị, file, các hệ thống file (filesystems), hoặc toàn bộ máy mà những người khác không thuộc nhóm có thể bị hạn chế.

Nhiều hệ Linux chỉ có một nhóm, nhóm mặc định, đây là cách đơn giản nhất để quản lý hệ thống. Trong trường hợp này, truy nhập của người dùng đến các thiết bị và file được điều khiển bằng quyền trên file chứ không phải bằng nhóm.

Các thông tin về nhóm được để trong file `/etc/group` có cấu trúc tương tự như file `/etc/passwd`. Đây là một file `/etc/group` được tạo khi mới cài đặt Linux

```
root::0:root
bin::1:root,bin,daemon
daemon::2:root,bin,daemon
sys::3:root,bin,adm
adm::4:root,adm,daemon
tty::5:
disk::6:root,adm
lp::7:lp
mem::8:
kmem::9:
wheel::10:root
floppy::11:root
mail::12:mail
news::13:news
uucp::14:uucp
man::15:man
users::100:games
nogroup::-1:
```


Các dòng có khuôn dạng như dưới đây, các trường phân cách bởi dấu ':', hai dấu ':' cạnh nhau thể hiện trường tương ứng rỗng.

```
group name:group password:group ID:users
```

Trong đó các trường có ý nghĩa như sau :

- *group name* : tên duy nhất xác định một nhóm thường gồm nhiều nhất là 8 kí tự.
- *password* : trường mật mã đã mã hoá thường để trống hoặc là dấu *, cũng có thể để chỉ mật mã mà một người muốn ra nhập nhóm phải nhập vào. Không phải mọi phiên bản của Linux hay Unix sử dụng trường này do đó nó được bỏ trống để có thể tương thích ngược
- *group ID (GID)* : là số duy nhất cho mỗi nhóm, được sử dụng bởi hệ điều hành.
- *Users* : chứa danh sách mọi tên người dùng (user name) thuộc nhóm đó, phân cách bởi các dấu ' ; '. Danh sách này không kể những người dùng thuộc nhóm đó theo số hiệu nhóm đã được ghi trong file /etc/passwd của người đó (tức là những thành viên mặc định của nhóm)

Mọi hệ Linux đều có một số các nhóm mặc định thuộc hệ điều hành, các nhóm này thường là bin, mailm uucp, sys Trừ hai nhóm cuối các nhóm còn lại đều là các nhóm phụ thuộc hệ thống. Do vậy không nên cho một người sử dụng thuộc vào các nhóm này vì chúng sẽ cho họ quyền tương tự như root. Chỉ có các đăng nhập hệ thống (vào hệ thống theo các tài khoản của hệ thống) mới cho phép truy nhập đến các nhóm của hệ điều hành.

4.3.3.1 Các nhóm mặc định của hệ thống

Các nhóm dùng để đặt quyền trên file và quyền truy nhập các tiện ích. Chúng ta xem qua các nhóm quan trọng nhất và chức năng của chúng.

- Nhóm root/wheel/system : thường dùng để cho phép người dùng sử dụng lệnh su để truy nhập dưới quyền root. Nhóm này làm chủ hầu hết các file
- Nhóm daemon : dùng để chỉ những người làm chủ thư mục spool (mail, máy in...)
- Nhóm kmem : được dùng cho các chương trình cần truy nhập hạt nhân, bộ nhớ trực tiếp (bao gồm cả lệnh ps)
- Nhóm sys làm chủ file hệ thống. Trong một số hệ thống nó có vai trò như kmem.
- Nhóm tty làm chủ tất cả các file đặc biệt làm việc với terminal.

Nhóm mặc định của phiên bản SlackWare của Linux nêu trên có nhóm mặc định là `users` có GID là 100. Các phiên bản khác có nhóm mặc định tên là `group`, cũng là chuẩn của hầu hết các hệ thống Linux.

4.3.3.2 Thêm nhóm

Để thêm một nhóm, ta có thể thao tác trực tiếp trên file `/etc/group` bằng tay sử dụng bất kì một trình soạn thảo ASCII nào, hoặc có thể dùng một tiện ích của shell như `addgroup` hoặc `groupadd`. Đa số các nhà quản trị hệ thống thích làm bằng tay hơn vì có thể nhìn thấy toàn bộ file `group` trong khi soạn thảo trên nó, hơn nữa không phải hệ thống Linux nào cũng có `addgroup` hay `groupadd`.

Để thêm một nhóm bằng tay, đầu tiên sao lưu file `/etc/group`. Dùng một trình soạn thảo ASCII và thêm một dòng cho mỗi nhóm mới muốn tạo, các dòng nhất thiết phải tuân theo cú pháp của file như đã nêu trên, nếu không với dòng ghi sai sẽ không cho người dùng không thể thuộc về nhóm đó. Ví dụ, sau đây hai nhóm được tạo :

```
accounts::51:bill
scanner::52:yvonne
```

Cũng như số hiệu người dùng, số hiệu nhóm nên được gán tuần tự cho tiện, và cũng nên sắp xếp file theo thứ tự của số hiệu nhóm (hoặc tên nhóm) dù điều này không bắt buộc. Như trong ví dụ cùng với việc tạo nhóm mới 2 người dùng cũng được đưa vào mỗi nhóm. Để thêm nhiều người dùng vào mỗi nhóm ta sẽ xem trong phần tiếp.

Cuối cùng, kiểm tra lại quyền trên file và quyền sở hữu, nên để root làm chủ và có nhóm làm chủ là root (hoặc system tùy theo nhóm có GID bằng 0) để tránh người dùng khác viết vào file.

4.3.3.3 Thêm người dùng vào một nhóm mới

Mỗi người dùng có thể thuộc nhiều nhóm mà tên người dùng (user name) của họ được ghi trong dòng tương ứng với nhóm đó trong file `/etc/group`. Tên người dùng trong một dòng được phân cách bằng dấu ',', thứ tự người dùng trong mỗi dòng không quan trọng. Ví dụ một vài dòng trong `/etc/group` tương ứng với nhóm có nhiều người dùng:

```
accounts::52:bill,yvonne,tim,roy,root
prgming:53:bill,tim,walter,gita,phyliss,john,root
cad:54:john,doreen,root
scanner:55:john,root,tim
```

Một khi đã đăng nhập vào hệ thống người dùng chỉ có thể thuộc về một nhóm tại một thời điểm, ban đầu sau khi đăng nhập, họ thuộc nhóm có số hiệu được ghi trong trường GID trong file `/etc/passwd`. Để chuyển đổi giữa các nhóm mà họ là thành viên, phải dùng lệnh `newgrp`.

4.3.3.4 Xóa một nhóm

Để xóa hoàn toàn một nhóm, hãy xóa dòng tương ứng nhóm đó khỏi `/etc/group`. Đồng thời kiểm tra file `/etc/passwd` xem có người dùng nào lấy nhóm vừa xóa làm nhóm khởi đầu khi đăng nhập hệ thống không, nếu có chuyển họ sang một nhóm mà họ là thành viên. Nếu ta không chuyển nhóm cho họ (tức là đổi trường GID trong `/etc/passwd` của họ) thì người dùng này không thể đăng nhập hệ thống được nữa vì có nhóm không hợp lệ. Thêm vào đó chúng ta phải kiểm tra toàn bộ hệ thống file xem có file hay thư mục nào lấy nhóm này làm chủ sở hữu hay không, nếu có đổi nó sang nhóm khác.

Một số phiên bản của Linux có các script của shell cho phép xóa các dòng trong `/etc/group`, các tiện ích này có tên `delgroup` hay `groupdel`, nhưng hầu hết các phiên bản của Unix không dùng chúng.

4.3.4 Lệnh su (switch user)

Đôi khi chúng ta muốn thực hiện một số lệnh như một người dùng khác. Nếu đang đăng nhập dưới tên siêu người dùng và muốn tạo file và đặt các quyền trên file và quyền sở hữu của bill thì đăng nhập lại dưới tên bill sẽ dễ làm việc hơn dưới tên root rồi đổi lại các tham số. Tương tự nếu người sử dụng bình thường muốn trở thành siêu người dùng trong một khoảng thời gian, họ phải thoát khỏi phiên làm việc, đăng nhập lại dưới tên mới và làm các thay đổi mong muốn. Để tránh những công việc đó, ta dùng lệnh `su`.

Lệnh `su` chuyển người dùng sang tên người dùng mới và trao cho họ các quyền mà người dùng mới có. Lệnh `su` lấy tên người dùng làm tham số, ví dụ khi người sử dụng đăng nhập dưới tên một người dùng bình thường và muốn có quyền sử dụng hệ thống như root họ có thể thực hiện câu lệnh sau :

```
$ su root
```

hoặc

```
$ su
```

khi đó Linux sẽ yêu cầu nhập mật mã của root, nếu gõ đúng, người sử dụng sẽ trở thành root cho đến khi nhấn Ctrl+D để thoát khỏi tài khoản này và trở về với tài khoản ban đầu.

Tương tự, nếu đang sử dụng hệ thống với account root mà người sử dụng muốn chuyển sang một người dùng khác ví dụ tparker, khi đó có thể thực hiện lệnh :

```
$ su tparker
```

Linux không yêu cầu người sử dụng nhập mật mã khi chuyển từ root sang vì họ đang có quyền của siêu người dùng.. Nếu đang đăng nhập dưới tên một người dùng bình thường khác root mà muốn chuyển sang một người dùng khác thì cần phải nhập mật mã.

Như vậy có thể thấy /etc/passwd và /etc/group là hai file cơ bản liên quan đến truy nhập Linux. Quản trị viên hệ thống có thể dễ dàng sửa đổi hai file này để thêm người dùng hay nhóm bất kì lúc nào. Nhưng cần ghi nhớ rằng đây là hai file sống còn và cần kiểm tra lại quyền truy nhập vào chúng sau khi sửa.

4.4 SAO LƯU VÀ KHÔI PHỤC

Khi chạy một hệ thống nhiều người dùng, có truy nhập mạng, email,... sao lưu sẽ là một khía cạnh rất quan trọng hàng ngày. Chỉ đến khi đã mất các thông tin quan trọng, các cấu hình nhiều người mới nhận ra được vai trò của sao lưu.

4.4.1 Tại sao phải sao lưu

Sao lưu là thực hiện sao một hệ thống file hoặc một phần của một hệ thống file vào một thiết bị lưu trữ khác để sau đó có thể dùng cho việc khôi phục hệ thống cũ. Hầu hết các hệ thống UNIX dùng băng từ làm thiết bị sao lưu, ta cũng có thể dùng các đĩa mềm hoặc đĩa cứng thứ cấp có thể tháo ra được.

Có rất nhiều khả năng dẫn đến hỏng dữ liệu trên đĩa trong các hệ thống máy tính hiện đại, hỏng do lỗi phần cứng, do mất nguồn điện, do các lệnh gõ sai... Đối với Linux nguyên nhân hỏng là do tự hệ điều hành. Vì Linux là một hệ thống đa người dùng và là hệ điều hành đa nhiệm, do đó bất kì lúc nào cũng có nhiều file hệ thống được mở. Linux giữ rất nhiều thông tin trong bộ nhớ về trạng thái hiện tại của nó và của các hệ thống file. Các thông tin này phải được viết lên đĩa thường xuyên. Khi các tiến trình của CPU bị ngắt các file và bảng hệ thống trong bộ nhớ có thể bị mất, do đó các file trên đĩa có thể bị đặt trong trạng thái tạm, điều đó không phù hợp với trạng thái hệ thống file thực.

Với các nguy cơ hỏng hệ thống file như vậy, mà hầu như ta không thể điều khiển được chúng, giải pháp sao lưu đôi khi là duy nhất để lấy lại các thông tin đã mất. Điều này có thể làm được dễ dàng nhờ các tiện ích như cron.

Một vấn đề phải quan tâm khi sao lưu là thiết bị dùng để sao lưu. Thông thường đó là đĩa mềm, băng từ, đĩa cứng... Cần đảm bảo rằng chúng được giữ xa các vùng có từ trường như dây điện thoại, modem, tivi, ... Thêm nữa là nên giữ chúng ở xa các máy chính, kiểu giữ các bản sao ở xa như thế này giúp cho ta có thể sử dụng chúng cho khôi phục trong các trường hợp xảy ra tai nạn như cháy, có thể làm hỏng cả hệ thống và thiết bị sao lưu.

4.4.2 Chọn thiết bị để sao lưu

Thông thường người ta hay dùng băng từ để sao lưu bởi các đặc tính giá thành thấp, các lưu trữ cũng khá đơn giản và tốc độ chấp nhận được. Quá trình đọc và ghi dữ liệu vào băng là đáng tin cậy, băng lại có thể chuyển từ máy này sang máy khác, tất nhiên là phải có ổ băng từ.

Cũng có thể dùng các thiết bị khác như ổ cứng có thể tháo lắp được dưới các dạng khác nhau như Iomega Bernoulli hay ổ ZIP, chúng đều dùng công nghệ đầu đọc từ như các ổ cứng bình thường. Nói chung thì giá của loại thiết bị này có thể cạnh tranh được với ổ băng từ.

Một số hệ thống băng từ quang học cho DOS và Windows cũng sử dụng dưới Linux. Chúng có hướng tới các băng có kích thước nhỏ khoảng 3.5 inch vừa với một ổ đĩa nhỏ. Chúng có ưu điểm nổi bật là không nhạy cảm đối với từ trường và có thể giữ được lâu. Loại băng nhỏ khoảng 230M có thể rẻ hơn cả một ổ băng từ, tuy vậy loại lớn 2.4G có giá xấp xỉ giá máy tính.

Đương nhiên ta cũng có thể dùng ổ cứng khác để sao lưu.

Một loại thiết bị khác là ổ CD-ROM ghi được và WORM (ghi một lần đọc nhiều lần) cũng có thể dùng được, tuy vậy chúng chỉ có thể ghi được một lần, do đó lí tưởng để lưu các thông tin vĩnh viễn.

Ổ đĩa mềm được xếp vào hàng cuối cùng trong các thiết bị sao lưu cho các hệ thống file lớn, mặc dù nó tốt cho các hệ thống file nhỏ.

4.4.3 Đặt kế hoạch sao lưu

Một vấn đề quan trọng của sao lưu là sao lưu thường xuyên. Điều này đặc biệt quan trọng đối với các hệ thống đa người dùng và thường xuyên thay đổi hệ thống file. Ta cũng không cần phải sao lưu mọi thứ mà chỉ cần sao lưu những file đã thay đổi kể từ lần sao lưu trước.

Các nhà quản trị hệ thống UNIX thích thực hiện sao lưu vào ban đêm vì lúc đó có ít người dùng đăng nhập, CPU không phải làm việc nặng nhọc, và hệ thống có ít file đang mở nhất. Nói chung việc sao lưu ít khi được thực hiện bằng tay mà được làm tự động nhờ chương trình cron, do đó có thể đặt thời gian thực hiện chính

xác để làm ảnh hưởng ít nhất đến các tiến trình nền mà hệ thống đang chạy. Tôt nhiên nếu hệ thống Linux của chúng ta không chạy thường xuyên mà chỉ chạy khi ta cần dùng thì cũng không cần sao lưu tự động mà hãy thực hiện khi ta đang làm các công việc khác trong trên thống.

Không bao giờ chỉ giữ đúng một bản sao của hệ thống vì như vậy ta sẽ không thể quay lui đến bản sao trước được, ví dụ ta thực hiện sao lưu hàng ngày nhưng lại muốn khôi phục lại một file đã xóa từ 1 tuần trước, khi đó điều này trở nên không thể được.

Lí tưởng nhất là chúng ta giữ các bản sao nhiều ngày, thậm trí hàng tuần trước khi dùng lại chúng. Trong các hệ thống nhiều người dùng điều này đặc biệt quan trọng vì đôi khi họ chỉ nhớ ra là cần một file sau khi đã xóa nó 2 tháng, khi đó ta đã quay vòng sử dụng băng vài lần. Kế hoạch lí tưởng phụ thuộc vào quan điểm của quản trị viên đối với việc sao lưu, nhưng một hệ thống sao lưu toàn diện yêu cầu ít nhất các bản sao lưu thặng dư hàng ngày (chỉ sao các phần thay đổi) của 2 tuần và một bản sao lưu toàn bộ mỗi tuần.

Sao lưu toàn bộ đòi hỏi phải sao tất cả các file của hệ thống do đó thiết bị lưu phải có kích thước bằng tổng kích thước hệ thống file. Có thể ta phải dùng nhiều thiết bị lưu cùng loại cho một lần sao lưu nếu dung tích của thiết bị nhỏ hơn hệ thống file cần lưu. Một số hệ thống sao lưu dùng các thuật toán nén do đó kích thước cần lưu thực sự sẽ nhỏ đi.

Sao lưu thặng dư (còn gọi là sao lưu chênh lệch) chỉ lưu các file đã bị thay đổi hoặc được tạo sau lần sao lưu gần nhất. Để xác định các file này ta có thể căn cứ vào ngày sửa file.

Sao lưu thặng dư đôi khi là khó khăn đối với Linux trừ phi ta có thể hạn chế một vùng của hệ thống file có vẻ như đã bị thay đổi để sao lưu. Sao lưu kiểu này gọi là sao lưu bộ phận. Ví dụ sao lưu hệ thống file /usr nếu mọi file của người dùng đều nằm ở đây.

Nên gán nhãn cho các băng sao lưu với các tên phản ánh cách ta dùng chúng. Ví dụ Ngày1, Ngày2 ... Ngày10. Sau đó khi phải quay vòng ta lại dùng lại Ngày1 (do đó Ngày1 luôn sau Ngày10).

Việc sao lưu bao nhiêu lâu một lần, như đã nói, là tùy thuộc vào từng hệ thống. Nói chung với các hệ thống đa người dùng, hay thay đổi thì nên thực hiện hàng ngày. Nên thực hiện một sao lưu toàn bộ sau bốn hay năm lần sao lưu bộ phận.

Một kế hoạch sao lưu mở rộng của sao lưu hàng ngày được nhiều quản trị viên thích dùng là chu kì sao lưu hàng ngày và hàng tuần. Ta chia hệ thống băng dùng để sao lưu thành 2 nhóm cho sử dụng hàng ngày và hàng tuần. Ví dụ, có 14 băng, ta dùng 10 băng cho sao lưu hàng ngày trong 2 tuần (trừ đi 4 ngày nghỉ cuối tuần của 2 tuần). Gán cho chúng các nhãn từ Ngày1 đến Ngày 10. Dùng 4 băng còn lại cho chu kì 2 tuần một và gọi chúng là Tuần 1,... Tuần 4. Như vậy hàng ngày sẽ

lần lượt dùng các băng Ngày, đến hết một vòng sử dụng các băng ngày thì dùng một băng Tuần. Sau đó lại quay vòng dùng các băng Ngày rồi đến băng Tuần...

Chu kì sao lưu này có những ưu điểm hơn hẳn chu kì sao lưu hàng ngày đơn giản. Với 10 băng sao Ngày ta có thể khôi phục lại thông tin trong vòng 2 tuần trước. Với 4 băng sao lưu hai tuần một (băng Tuần) ta có thể khôi phục một file từ cách đó 8 tuần tức là một file hoặc một nhóm file từ 2 tháng trước chứ không phải chỉ hai tuần. Với nhiều băng hơn, ta có thể mở rộng chu kì sao lưu hoặc thêm chu kì sao lưu hàng tháng.

4.4.4 Giữ các log của sao lưu

Sau khi đã sao lưu dữ liệu vài lần, đến lúc cần khôi phục một file, lúc đó chúng ta không biết rằng file đó nằm trong băng lưu nào hay băng đó lưu lại ngày nào. Để giải quyết vấn đề này, một số người ghi ngày, nội dung của băng lên một mảnh giấy gắn vào băng. Cách này không hay lắm vì ta phải lục tất cả các băng xem file ta cần ở đâu. Do đó chúng ta cần giữ các log sao lưu.

Mỗi khi sao lưu, ta lại cập nhật log sao lưu. Log sao lưu chỉ đơn giản là một tờ giấy hoặc một quyển sổ có các cột chứa các thông tin cơ bản sau :

- Ngày sao lưu,
- Tên băng sao lưu (ví dụ Ngày 1),
- Hệ thống file được sao lưu,
- Sao lưu toàn bộ hay bộ phận, nếu sao lưu bộ phận thì sao lưu thư mục nào.

Các thông tin này chỉ cần ghi lại trong vài giây mà thôi. Với các hệ thống lớn, ta có thể thêm một số thông tin khác cho đầy đủ hơn.

- Ai sao lưu,
- Sao lưu bằng tay hay tự động,
- Nơi cất băng lưu.

Ngày sao lưu giúp ta theo dõi thời điểm sao lưu lần cuối, đồng thời cũng là chỉ số để khôi phục file. Nếu một người dùng của hệ thống muốn khôi phục một file mà anh ta biết đã xoá nhầm 2 tuần trước, ta có thể căn cứ vào ngày để tìm ra băng chứa file đó nhờ vào ngày của log sao lưu.

Để thuận tiện nên giữ log sao lưu ở gần hệ thống, có người thích để nó ở cùng với các băng sao lưu.

4.4.5 Dùng tar để sao lưu

Chương trình tar (tape archiver : lưu trữ băng) là lệnh lưu các file và thư mục vào một thiết bị lưu trữ và khôi phục chúng sau đó. Lệnh tar tạo một file lưu trữ, là file duy nhất chứa nhiều file trong nó (giống PKZIP làm trong DOS). Lệnh tar chỉ làm việc với các file mà nó tạo ra.

Khuôn dạng của lệnh tar :

tar switch modifiers files

Phần files của lệnh chỉ các file hay thư mục người sử dụng muốn lưu hay khôi phục. Người sử dụng có thể lưu toàn bộ hệ thống file /usr và bởi vậy khi khôi phục người sử dụng có thể chỉ cần khôi phục một file, ví dụ /usr/tparker/big_file

Tham số switch chỉ cách tar ghi và đọc từ thiết bị lưu. Người sử dụng chỉ có thể dùng một switch với tar tại một thời điểm. Các switch có thể là:

- c Tạo một file lưu mới
- r Ghi vào cuối file lưu hiện tại
- t Liệt kê danh sách các file trong file lưu
- u Thêm file chưa bị sửa hoặc chưa được lưu
- x Trích ra từ file lưu

Cũng có thể thêm một số các modifiers để điều khiển file lưu và cách tar dùng nó. Sau đây là các modifier:

- A Bỏ tên file tuyệt đối (gồm cả đường dẫn)
- b Đưa ra tham số khối (0-20)
- e Chống cất file để ghi trên các thiết bị lưu khác nhau
- f Chỉ ra tên thiết bị lưu trữ
- F Specifies the name of a file for tar arguments
- k Đưa ra kích thước của thiết bị lưu (kilobytes)
- l Đưa ra thông báo lỗi nếu không thể liên kết được
- m Không khôi phục thời điểm sửa
- n Chỉ ra thiết bị lưu không phải là băng
- p Trích ra các file với các quyền của nó như ban đầu
- v Liệt kê danh sách file
- w Hiện thị hành động lưu và chờ người dùng khẳng định lại.

Lệnh tar dùng đường dẫn tuyệt đối cho hầu hết các hoạt động trừ khi có modifier A.

4.4.5.1 Sao lưu bằng đĩa mềm

Cách dễ nhất để sao lưu hệ thống bằng đĩa mềm là dùng lệnh *tar*. Lệnh:

```
# tar cvfzM /dev/fd0 /
```

sẽ tạo ra một bản sao hệ thống đầy đủ sử dụng ổ đĩa mềm /dev/fd0. lựa chọn "M" của tar cho phép tạo bản sao bằng nhiều đĩa mềm. Để khôi phục lại hệ thống bằng bản sao, bạn gõ lệnh:

```
# tar xvfzM /dev/fd0
```

Bạn cũng có thể sử dụng cách này đối với băng từ.

Tạo một bản sao hệ thống đầy đủ bằng đĩa mềm sẽ tốn rất nhiều thời gian và tài nguyên. Nhiều quản trị viên hệ thống sử dụng phương pháp sao lưu tăng dần (incremental backup). Mỗi tháng một bản sao đầy đủ được tạo ra và mỗi tuần chỉ những file đã thay đổi tuần trước mới được lưu lại.

4.4.5.2 Sao lưu bằng ổ Zip

Việc sao lưu bằng ổ Zip cũng gần giống với việc sao lưu bằng ổ mềm tuy nhiên dung lượng của các đĩa Zip lớn hơn các đĩa mềm rất nhiều.

Một ổ đĩa Zip khi được mount là /dev/sda4:

```
# mount /dev/sda4 /mnt
```

Để tiện dụng hơn khi sử dụng các ổ Zip, bạn tạo một thư mục riêng cho ổ Zip:

```
# mkdir /zip
# chmod 0755 /zip
```

Ví dụ để nén và lưu trữ thư mục /etc vào ổ Zip, bạn sử dụng lệnh:

```
# tar zcvf /zip/etc.tgz /etc
```

Bây giờ để giải nén, bạn sử dụng lệnh:

```
# cd /  
# tar zxvf /zip/etc.tgz
```

4.4.5.3 Sao lưu bằng băng từ

Nếu bạn dùng một ổ băng là /dev/tape để lưu toàn bộ hệ thống file, bạn dùng lệnh :

```
# tar cvf /dev/tape /
```

Lựa chọn f cho phép ta chỉ ra tên thiết bị /dev/tape. Toàn bộ hệ thống file được lưu trong file lưu mới tạo (chỉ ra bởi lựa chọn c). Mọi thứ đang có trong băng sẽ bị ghi đè khi file lưu mới được tạo. Nếu thêm lựa chọn v trong lệnh, tar sẽ hiển thị tên file và các kích thước của chúng lên màn hình sau khi đã lưu.

Để khôi phục toàn bộ hệ thống file từ một băng đã dùng trong ví dụ trước, dùng lệnh :

```
# tar xvf /dev/tape
```

Để khôi phục một file trong băng có thể chỉ rõ file đó :

```
# tar xvf /dev/tape /usr/tparker/big_file
```

lệnh này chỉ khôi phục file /usr/tparker/big_file.

Để lấy danh sách tất cả các file trong băng lưu, ta dùng lệnh :

```
# tar tvf /dev/tape
```

Lựa chọn v cho phép hiển thị danh sách lên màn hình.

Hầu hết các băng đều đòi hỏi phải đưa ra hệ số khối khi tạo một file lưu, nhưng ta không cần phải chỉ ra số này khi đọc băng vì tar có thể tự nhận ra. Hệ số khối báo cho tar biết lượng dữ liệu viết vào một khúc trong băng. Khi lưu vào băng, bạn chỉ ra hệ số này với modifier b. Ví dụ :

```
# tar cvfb /dev/tape 20 /usr
```

Lệnh này tạo một file lưu mới trên /dev/tape có hệ số khối 20 và chứa tất cả các file của /usr (nói chung mọi băng đều hoạt động với hệ số khối 20). Chú ý rằng các tham số modifier phải đúng thứ tự với kiểu modifier đã chỉ ra, như ở đây f trước b do đó tên thiết bị phải được chỉ ra trước hệ số khối.

Một vấn đề khác là băng có thể không đủ chứa toàn bộ hệ thống file, do đó phải dùng nhiều băng. Lựa chọn k được sử dụng để báo cho tar biết kích thước của băng, nó sẽ tương ứng với tham số là dung lượng của băng theo kilobyte. Ví dụ :

```
# tar cvbfk 20 /dev/tape 122880 /usr
```

Như vậy ta dùng hệ số khối là 20 cho thiết bị lưu /dev/tape, dung lượng băng là 122880KB.

4.5 THAY ĐỔI CẤU HÌNH HẠT NHÂN

Nhìn chung, ta không nên động đến hạt nhân Linux trừ phi cần thực hiện các công việc nâng cấp, cài đặt mạng (như NFS hay NIS), cài đặt các trình điều khiển thiết bị mới có các yêu cầu đặc biệt về hạt nhân. Các quá trình cài đặt này thường được thực hiện bằng phần mềm, tuy vậy trong phần này ta cũng xem xét quá trình làm việc với hạt nhân nói chung. Cần lưu ý rằng thay đổi cấu hình hạt nhân là một công việc mạo hiểm, mọi sai sót trong việc sử mã nguồn có thể làm ảnh hưởng đến toàn bộ hệ thống file. Vì vậy mà trước khi làm việc với hạt nhân hãy tạo các đĩa khởi động và sao lưu hệ thống để tránh những sai sót đáng tiếc không thể phục hồi được xảy ra.

Các phiên bản Linux nói chung thường không hoàn toàn đồng nhất. Vì vậy các lệnh sau đây có thể không đúng với một số phiên bản Linux, nhưng cách tiếp cận vấn đề là như nhau, có thể chỉ có tên thư mục và tên tiện ích là khác nhau.

Hạt nhân Linux được dịch bằng chương trình dịch C (là một phần của Linux), vì vậy trong phần này ta cũng xem xét đôi chút về trình dịch C, những gì có thể liên quan đến việc thay đổi cấu hình hạt nhân.

4.5.1 Nâng cấp và cài đặt phần mềm hạt nhân mới

Linux là một hệ điều hành linh hoạt. Các phiên bản mới của hạt nhân hay các phần của hệ điều hành có thể được liên kết vào hạt nhân luôn sẵn sàng cho mọi người dùng. Ta có thể sẽ phải biên dịch lại hạt nhân khi thêm phần mềm mới, hoặc không chỉ khi phần mềm đó được nạp dưới dạng tiện ích hoặc trình điều khiển thiết bị.

Nói chung nên tránh thường xuyên cập nhật hệ thống với mọi phiên bản mới, vì như vậy có thể xảy ra vấn đề không tương thích với hệ thống hiện tại, hoặc phải

thường xuyên cấu hình lại hệ thống do việc cập nhật hay xoá mất các thông tin cấu hình. Việc cập nhật khiến ta phải tiêu phí thời gian vào việc dịch lại hạt nhân, do đó nên cân nhắc xem phiên bản đó có đáng giá với thời gian cài đặt và các vấn đề có thể nảy sinh hay không. Nói chung chỉ nên nâng cấp một đến hai lần trong một năm thôi.

Ta cũng nên nghĩ rằng khi nâng cấp cũng không bắt buộc phải nâng cấp mọi thứ. Nhiều phiên bản mới của Linux chỉ thay đổi khoảng 5% so với hệ thống cũ và chỉ các module chính mới được nâng cấp. Do vậy cũng chỉ nên nâng cấp những phần có ảnh hưởng nhất định như chương trình dịch, thư viện, các tiện ích thường dùng. Phương pháp này tiết kiệm thời gian và việc cấu hình lại.

4.5.2 Dịch lại hạt nhân từ mã nguồn

Mọi việc thay đổi cấu hình hạt nhân : nâng cấp, thay thế, thêm mã mới vào hạt nhân đều được thực hiện theo cách : lấy mã nguồn của hạt nhân, thay đổi cấu hình, dịch lại và đặt nó vào đúng vị trí trong hệ thống file để hệ thống được chạy một cách đúng đắn (nói chung quá trình này đều được tự động bằng các chương trình).

4.5.2.1 Tìm mã nguồn hạt nhân

Các chương trình nguồn hạt nhân Linux thường có trên CD-ROM, các địa chỉ FTP, các nhóm người dùng,... Hầu hết các phiên bản hạt nhân được đánh số bằng tên phiên bản và một mức patch. Ví dụ một hạt nhân tên là 1.12.123, trong đó 1 là số phiên bản chính, 12 là số phiên bản phụ, 123 là số patch. Hầu hết các nơi có các chương trình nguồn hạt nhân đều chứa vài phiên bản một lúc.

Đa số các chương trình nguồn hạt nhân đều được đặt dưới một file nén. Giải nén file vào một thư mục con của /usr/src, đây cũng là thư mục chứa hầu hết các chương trình nguồn của Linux.

4.5.2.2 Dùng các chương trình nguồn hạt nhân mới

Với chương trình nguồn hạt nhân mới, nhiệm vụ của ta là tải chương trình này, sau đó dịch và cấu hình lại.

Để giải nén chương trình này vào thư mục /usr/src, ta phải tạo thư mục /usr/src/linux, nó sẽ đè lên chương trình nguồn hạt nhân cũ. Để an toàn, trước tiên ta nên lưu lại chương trình hạt nhân cũ này để có một bản sao lưu phòng bất chắc.

Sau đó tạo 2 liên kết biểu tượng đến thư mục /usr/include. Thường dùng lệnh sau :

```
# ln -sf /usr/src/linux/include/linux
/usr/include/linux
```

```
# ln -sf /usr/src/linux/include/asm
/usr/include/asm
```

nếu không có 2 liên kết này quá trình nâng cấp sẽ không thực hiện được.

Sau khi đã tời chương trình nguồn và đặt các liên kết biểu tượng (symbolic link), ta có thể bắt đầu quá trình dịch. Để dịch ta dùng gcc hay g++ (chương trình dịch GNU C và C++) hoặc các chương trình tương thích khác, nên xem xem chương trình nguồn mới có được hỗ trợ bởi gcc hay g++ không.

Tìm file Makefile (thường trong thư mục /usr/src/linux), file này thường có một dòng định nghĩa ROOT_DEV, chỉ ra hệ thống file gốc khi Linux khởi động. Thường là

```
ROOT_DEV = CURRENT
```

Nếu ta có các giá trị khác, cần đảm bảo là nó đúng với cấu hình hệ thống file. Nếu Makefile không có giá trị, hãy đặt cho nó như dòng trên.

Quá trình dịch bắt đầu bằng việc chuyển đến thư mục /usr/src/linux và đưa ra lệnh :

```
# make config
```

nó gọi tiện ích make của chương trình dịch C. chương trình config đưa ra một loạt các câu hỏi cấu hình yêu cầu ta trả lời trước khi dịch, đó là các câu hỏi về kiểu đĩa đang dùng, CPU, các phân hoạch, hay các thiết bị khác như CD-ROM. Nếu ta không biết chắc các thông tin này hãy dùng giá trị mặc định. Trường hợp xấu nhất là ta phải làm lại thao tác này.

Tiếp theo là đặt các phụ thuộc nguồn (source dependencies). Dùng lệnh sau :

```
# make dep
```

Nếu phần mềm mà ta đang cài đặt không có file dep, kiểm tra lại hướng dẫn cài đặt xem các phụ thuộc có phải được đặt trong các bước khác hay không.

Cuối cùng ta dịch hạt nhân mới. Dùng lệnh :

```
# make Image
```

nó sẽ dịch hạt mã nguồn nhân mới và đặt file ảnh hạt nhân trong thư mục hiện tại (như ở đây là `/usr/src/linux`). Nếu muốn có một ảnh nén hạt nhân, ta dùng lệnh :

```
# make zImage
```

tuy vậy không phải phiên bản nào cũng hỗ trợ dịch ảnh nén.

Bước cuối cùng, sao file ảnh hạt nhân mới vào đĩa khởi động hay đĩa mềm khởi động. Để đặt vào đĩa mềm khởi động dùng lệnh :

```
# cp Image /dev/fd0
```

Bằng cách khác, nếu muốn dùng LILO để khởi động hệ điều hành, ta cài đặt hạt nhân mới bằng cách chạy chương trình cài đặt hoặc `/usr/lilo/lilo`. Nhưng đừng chép đè hạt nhân mới lên hạt nhân cũ của đĩa khởi động.

Còn lại chỉ là khởi động lại hệ thống. Nếu có vấn đề gì, hãy khởi động lại bằng đĩa mềm và khôi phục lại hạt nhân cũ.

4.5.3 Thêm các trình điều khiển thiết bị vào hạt nhân

Đôi khi ta muốn liên kết một trình điều khiển thiết bị mới (chẳng hạn khi thêm một ổ đĩa quang vào hệ thống) hay một phần mềm nào đó vào hạt nhân mà không phải thực hiện quá trình nâng cấp hạt nhân. Các phần mềm ghép hạt nhân (add-in kernel) thường có các hướng dẫn cài đặt, nhưng quá trình nói chung là :

- Đặt mã nguồn vào trong thư mục mà tiến trình dịch lại hạt nhân có thể tìm được (thư mục `/usr/src`).
- Sửa file *Makefile* để yêu cầu tạo thêm mã mới vào hạt nhân. Việc này được thực hiện bởi script cài đặt hoặc ta tự làm lấy. Một số phần mềm có file *Makefile* riêng phục vụ việc này.

Sau đó ta bắt đầu dịch lại hạt nhân, tương tự như đã chỉ ra trong phần trên.

4.5.4 Nâng cấp thư viện

Hầu hết các phần mềm trên hệ thống Linux đều dùng các thư viện chia sẻ. Đôi khi sau khi nâng cấp hệ thống chúng ta gặp phải thông báo:

```
Incompatible library version
```

điều đó có nghĩa là thư viện đã được nâng cấp nhưng chưa được dịch lại. Hầu hết các thư viện đều tương thích ngược, vì vậy các phần mềm hiện tại sẽ làm việc tốt sau khi nâng cấp thư viện.

Thư viện ít thường xuyên được nâng cấp hơn là hạt nhân, các chương trình thư viện nâng cấp cũng hay đặt cùng chỗ với các chương trình hạt nhân nâng cấp. Thường thì có các tài liệu hướng dẫn chỉ ra đâu là thư viện mới nhất, hoặc các thư viện nào là cần thiết cho phiên bản mới của hạt nhân. Các bản nâng cấp thư viện cũng là các file nén, qua trình thì chúng cũng giống như trình nguồn hạt nhân, chỉ khác thư mục đích là `/lib`, `/usr/lib`, và `/usr/include`. Thường các file `.a` hay `.aa` sẽ vào thư mục `/usr/lib`, các file `libc.so.version` vào `/lib`.

Ta phải chuyển các liên kết biểu tượng của hệ thống file để trở đến phiên bản mới nhất của thư viện, ví dụ ta đang chạy phiên bản thư viện `libc.so.4.4.1` và nâng cấp lên `libc.so.4.4.2`, ta phải thay liên kết biểu tượng đặt trong `/lib` đến file này.

```
# ln -sf /lib/libc.so/4/4/1 /lib/libc.so.4
```

Trong đó tham số cuối cùng của lệnh trên là tên file thư viện hiện tại trong `lib`.

Ta cũng phải đổi liên kết biểu tượng đến file `libc.so.version` theo cách đó. Lưu ý là không được xóa các liên kết biểu tượng, nếu không tất cả các chương trình phụ thuộc vào thư viện (gồm cả `ls`) sẽ không hoạt động được.

4.5.5 Sử dụng chương trình dịch C

Để dịch hạt nhân và hầu hết các tiện ích, Linux dùng chương trình dịch C. Chương trình dịch C có sẵn đối với mọi phiên bản của Linux là trình dịch GNU C gọi tắt là `gcc`, nó tương thích hoàn toàn với ANSI C, nếu đã quen với một trình dịch C trên một hệ điều hành khác rồi, bạn sẽ học `gcc` rất nhanh.

Cú pháp cơ bản của `gcc` như sau:

```
gcc [options] [filenames]
```

trong đó *filenames* là tên các file cần dịch, *options* là các lựa chọn của lệnh `gcc` sẽ được áp dụng trên mọi file cần dịch. `gcc` có rất nhiều lựa chọn sau đây ta sẽ xem một số lựa chọn cần thiết.

4.5.5.1 Các lựa chọn của chương trình dịch

Nhiều lựa chọn của `gcc` có nhiều hơn một kí tự, do đó các lựa chọn này nhất thiết phải đặt sau dấu '-', và cũng vì vậy ta không thể gộp các lựa chọn như vẫn làm với các lệnh Linux khác. Ví dụ sau đây là 2 lệnh có ý nghĩa khác nhau:

```
# gcc -p -g test.c
# gcc -pg test.c
```

Khi dịch một chương trình bằng gcc mà không có lựa chọn nào trên dòng lệnh, ta sẽ thu được một file thực thi tên là a.out. Để đặt cho file thực thi tên khác, ta dùng lựa chọn -o. Tên file thực thi được đặt ngay sau lựa chọn -o. Ví dụ dịch file count.c thành file thực thi count :

```
# gcc -o count count.c
```

Các lựa chọn khác cho phép ta chỉ ra mức độ dịch mà ta muốn. Lựa chọn -c yêu cầu gcc dịch mã nguồn đến dạng object mà không đến giai đoạn dịch sang mã assembler và giai đoạn liên kết. Lựa chọn này thường được dùng vì nó làm cho quá trình dịch các chương trình C gồm nhiều file nhanh hơn và dễ quản lí. File object có đuôi mặc định là .o.

Lựa chọn -S yêu cầu gcc ngừng dịch sau khi đã sinh ra file assembler (mặc định đuôi file assembler là .s).

Lựa chọn -E yêu cầu chương trình dịch hiển thị các giai đoạn tiền xử lí dịch trên file đầu vào. Khi đặc lựa chọn này đầu ra của tiền xử lí được chuyển ra đầu ra chuẩn (mặc định là màn hình) chứ không được lưu trong file.

Khi dịch bằng gcc, gcc sẽ cố gắng dịch mã nguồn trong thời gian ngắn nhất, dễ gỡ rối nhất (tức là mã sau dịch có thứ tự như mã nguồn) mà không thực hiện một tối ưu hoá nào cả. Chương trình gcc cung cấp cho ta nhiều lựa chọn để yêu cầu dịch ra các chương trình thực thi nhỏ hơn, nhanh hơn, đó là các lựa chọn -O và -O2.

Lựa chọn -O yêu cầu gcc thực hiện các tối ưu cơ bản trên mã nguồn làm cho chương trình chạy nhanh hơn trong đa số các trường hợp. Lựa chọn -O2 yêu cầu tạo ra mã nhỏ nhất và chạy nhanh nhất có thể. Lựa chọn -O2 làm cho tốc độ dịch chậm hơn lựa chọn -O nhưng kết quả là mã dịch được chạy nhanh hơn.

4.5.5.2 Các lựa chọn gỡ rối và profile

Gcc cung cấp một số lựa chọn gỡ rối và profile. Hai trong số các lựa chọn này thường được dùng nhất là : -g và -pg.

Lựa chọn -g yêu cầu gcc sinh ra các thông tin gỡ rối để trình gỡ rối của GNU (gdb) dùng giúp ta gỡ rối chương trình. Với gcc ta có thể kết hợp cả hai lựa chọn -O và -g, khi đó ta vừa có thể sinh ra mã tối ưu, vừa có thể thực hiện gỡ rối mã gần nhất với mã được sinh ra cuối cùng. Khi dùng hai lựa chọn này kết hợp với nhau, có thể gcc sẽ thay đổi một chút mã chương trình mà ta đã viết khi thực hiện tối ưu mã.

Lựa chọn -pg yêu cầu gcc thêm mã mở rộng vào chương trình của chúng ta, khi đã thực thi các mã mở rộng này sẽ sinh ra các thông tin sơ lược mà chương trình gprof dùng để hiển thị thông tin về chương trình của ta.

4.5.5.3 Chương trình gỡ rối gdb

Linux có chương trình gỡ rối của GNU gọi là gdb, ta có thể dùng nó để gỡ rối chương trình C hay C++. Nó cho phép chúng ta thấy được cấu trúc bên trong hay bộ nhớ mà một chương trình dùng khi nó thực thi. Chương trình gỡ rối cho phép ta thực hiện các chức năng sau :

- Điều khiển các giá trị hay biến có trong chương trình.
- Đặt breakpoint để dừng chương trình tại một dòng lệnh xác định.
- Thực hiện từng dòng lệnh một.

Ta có thể dùng gdb với một số các lựa chọn trên dòng lệnh. Ta thường chạy gdb nhất với lệnh như sau :

```
gdb filename
```

Trong đó *filename* là tên file mà ta muốn thực hiện gỡ rối. Ngoài ra gdb còn cung cấp một số lựa chọn khác, ta có thể xem chi tiết các lựa chọn này bằng cách dùng lệnh man với gcc.

Để gdb làm việc đúng, ta cần dịch chương trình theo cách mà chương trình dịch sẽ sinh ra các thông tin gỡ rối, tức là dùng lựa chọn -g khi dịch. Thông tin gỡ rối được sinh ra bao gồm kiểu mỗi biến trong chương trình cũng như ánh xạ giữa các địa chỉ trong chương trình thực thi với các số dòng trong mã nguồn. Chương trình gdb sẽ dùng các thông tin này để liên hệ mã thực thi với mã nguồn.

4.6 TRONG TRƯỜNG HỢP KHẨN CẤP

Trong một số trường hợp, người quản trị hệ thống phải đối mặt với các vấn đề về hệ thống. Lời khuyên tốt nhất là : *hãy bình tĩnh*.

Bước đầu tiên để bạn có thể sửa chữa được vấn đề là phải tìm được vấn đề là gì. Bạn hãy xem hệ thống làm việc như thế nào khi có lỗi, trước khi bạn phải cầu cứu đến người khác.

Đôi khi, cần thiết phải cài đặt lại toàn bộ hệ thống từ đầu. Nhiều người mới sử dụng khi xóa nhầm một số file quan trọng của hệ thống, và vội tìm đến các đĩa cài đặt. Trong hầu hết các trường hợp như vậy, bạn có thể cứu lại hệ thống từ một đĩa bảo vệ.

4.6.1 Khôi phục bằng đĩa bảo vệ

Một trong những công cụ không thể thiếu được của người quản trị hệ thống được gọi là đĩa "boot/root", đó là một đĩa mềm có thể khởi động cho một hệ thống Linux đầy đủ, không phụ thuộc vào đĩa cứng. Việc tạo ra các đĩa boot/root rất đơn giản. Bạn tạo ra một hệ thống file root trên đĩa mềm, đặt mọi tiện ích cần thiết lên đó, sau đó bạn cài đặt LILO và hạt nhân hệ điều hành lên đĩa. Một cách khác là bạn có thể dùng một đĩa cho hạt nhân, một đĩa cho hệ thống file root. Cả hai cách đều cho cùng kết quả: bạn có thể chạy hệ thống Linux đầy đủ chỉ từ ổ đĩa mềm. Ví dụ về các đĩa boot/root là các đĩa khởi động Slackware hoặc H.J Lu.

Việc sử dụng một đĩa boot/root rất đơn giản. Bạn chỉ việc khởi động bằng đĩa này và đăng nhập với *root* (thông thường không có mật khẩu). Sau đó để có thể truy cập các file trên đĩa cứng, bạn phải *mount* lại hệ thống file bằng tay. Ví dụ:

```
# mount -t ext2 /dev/hda2 /mnt
```

Nhớ rằng / bây giờ nằm trên đĩa boot/root, do đó chẳng hạn thư mục */etc/passwd* trên đĩa cứng bây giờ sẽ là */mnt/etc/passwd* trên đĩa boot/root.

4.6.2 Sửa mật khẩu root

Nếu bạn quên mất mật khẩu của root, bạn có thể khởi động từ ổ mềm để sửa lại, sau đó bạn mount hệ thống file root vào /mnt, và xoá trường password trong file */mnt/etc/passwd* như sau:

```
root::0:0:root:/:/bin/sh
```

Bây giờ root sẽ không có mật khẩu nữa. Bạn có thể khởi động lại bằng đĩa cứng và đặt lại mật khẩu bằng lệnh *passwd*.

4.6.3 Hệ thống file bị rác

Nếu hệ thống file của bạn bị rác, bạn có thể chạy chương trình *e2fsck* hoặc dạng tương ứng của chương trình *fsck*. Trong hầu hết các trường hợp, đây là cách an toàn nhất để sửa lỗi dữ liệu trên đĩa cứng từ đĩa mềm.

Nguyên nhân thường gặp của hệ thống file bị rác là một *siêu block* bị hỏng. Một siêu block là phần đầu của hệ thống file chứa các thông tin như trạng thái, kích thước, các block trống, ... Nếu bạn bị hỏng siêu block, chẳng hạn có thể do bạn đã ghi nhầm dữ liệu trực tiếp lên bảng phân hoạch đĩa, hệ thống sẽ không thể nhận dạng được hệ thống file. Việc mount hệ thống file bị lỗi và chương trình *e2fsck* sẽ không thể sửa lỗi được.

May mắn thay, kiểu hệ thống file ext2fs đã lưu một bản sao của siêu block. Để *e2fsck* có thể sử dụng bản sao này, bạn sử dụng lệnh:

```
# e2fsck -b 8139 partition
```

trong đó *partition* là bảng phân hoạch đĩa bị hỏng. Tham số *-b 8139* có nghĩa là *e2fsck* sẽ sử dụng bản sao được lưu tại block 8139 trong hệ thống file.

4.6.4 Cứu các file đã mất

Nếu bạn vô tình xóa mất một file quan trọng của hệ thống, không có cách nào có thể khôi phục nó được. Tuy nhiên bạn có thể copy lại file này từ đĩa mềm vào đĩa cứng. Chẳng hạn nếu bạn xóa nhầm chương trình cho phép login: */bin/login*, bạn chỉ cần khởi động lại từ ổ mềm, mount hệ thống file root vào */mnt*, sau đó bạn sử dụng lệnh:

```
# cp -a /bin/login /mnt/bin/login
```

Tham số *-a* có nghĩa là lệnh này sẽ bảo toàn quyền truy nhập file được copy. Tuy nhiên nếu file bạn xóa không phải là các file cốt yếu đã được lưu trên đĩa boot/root, bạn sẽ không thể làm gì hơn. Cách tốt nhất là bạn nên lưu lại một bản sao đầy đủ của hệ thống.

Chương 5. SỬ DỤNG MẠNG TRONG LINUX

Linux là một hệ điều hành mạng, do vậy để có thể khai thác được hết các tính năng của hệ điều hành này trong chương này chúng ta sẽ tìm hiểu một số khái niệm cơ bản về mạng, giao thức mạng *TCP/IP* - một trong những giao thức cơ sở về mạng, và cách làm việc với mạng trong Linux.

5.1 GIỚI THIỆU CHUNG VỀ MẠNG TRONG LINUX

Linux cho phép các máy tính được nối lại với nhau thành một mạng để trao đổi thông tin và chia sẻ tài nguyên giữa các máy với nhau. Người sử dụng có thể đăng nhập vào hệ thống tại bất kỳ máy nào trong mạng giống như khi làm việc trên một máy duy nhất. Linux cũng hỗ trợ các giao thức cần thiết để bạn có thể kết nối với mạng Internet.

Internet - cũng được biết với tên gọi Net - là mạng máy tính lớn nhất thế giới, hoặc chính xác hơn là mạng của các mạng, tức bao gồm nhiều mạng máy tính được nối lại với nhau. Một số mạng máy tính bao gồm một máy tính trung tâm (còn gọi là máy chủ hay máy phục vụ) và nhiều máy khác (còn gọi là máy khách hàng hay trạm làm việc) nối vào nó. Các mạng khác, kể cả Internet, có quy mô lớn hơn, bao gồm nhiều máy chủ và cho phép bất kỳ một máy tính nào trong mạng có thể kết nối với bất kỳ máy nào khác để trao đổi thông tin thoải mái với nhau. Một khi đã được kết nối vào Internet, máy tính của bạn sẽ là một trong số hàng chục triệu thành viên của mạng khổng lồ này.

Mạng của các trường đại học và viện nghiên cứu là những thành viên lâu đời của Internet. Số người dùng gia nhập Internet tăng với tốc độ rất nhanh. Theo ước tính, con số này sẽ là 16 triệu vào cuối năm nay.

Giao thức được sử dụng để kết nối trên Internet là *TCP/IP*. Do vậy, để làm chủ được mạng máy tính và hoà mình vào thế giới Internet với Linux, bạn cần nắm được giao thức mạng *TCP/IP*.

5.2 THIẾT LẬP CẤU HÌNH MẠNG TCP/IP TRONG LINUX

5.2.1 Khái niệm mạng TCP/IP

TCP_IP (Transmission Control Protocol/Internet Protocol) là một giao thức được sử dụng rộng rãi trong các mạng internet lẫn các mạng cục bộ ethernet. Sau đây là mô hình kiến trúc mạng của TCP_IP khi so sánh với mô hình của OSI (Open Systems Interconnection):

OSI	TCP/IP
• <i>Tầng ứng dụng</i>: là tầng trên cùng trong mô hình mạng 7 tầng, nó cung cấp các dịch vụ cho các chương trình ứng dụng và làm rõ cách thức giao diện giữa các chương trình. • <i>Tầng trình diễn</i>: đảm nhận việc biểu diễn dữ liệu. • <i>Tầng phiên</i>: đảm nhận việc tạo, quản lí, và kết thúc các liên kết trên mạng. • <i>Tầng giao vận</i>: kiểm soát lỗi và các chuỗi dữ liệu truyền trên mạng. • <i>Tầng mạng</i>: đảm nhận việc đánh địa chỉ và chọn đường, kiểm soát luồng dữ liệu.	• <i>Tầng ứng dụng</i>: đảm nhận các chức năng của tầng ứng dụng, tầng trình diễn, và một phần của tầng phiên theo mô hình OSI. • <i>Tầng giao vận</i>: quản lí việc chọn đường và truyền dữ liệu tới đích (sử dụng giao thức TCP hoặc UDP-User Datagram Protocol) • <i>Tầng internet</i>: thực hiện việc đánh địa chỉ và truyền dữ liệu, phân đoạn và tổng hợp lại các gói dữ liệu (sử dụng giao thức IP).
• <i>Tầng liên kết</i>: xác định các phương pháp truy nhập đường truyền. • <i>Tầng vật lí</i>: làm rõ các đặc tính vật lí và sự hoạt động của đường truyền vật lí.	• <i>Tầng truy nhập mạng</i>: làm rõ các thủ tục truyền dữ liệu trên mạng, bao gồm cả các phương pháp truy nhập đường truyền.

5.2.2 Địa chỉ IP

Trong mạng TCP/IP, mỗi máy tính được định danh bởi một địa chỉ IP gồm 32 bits được viết dưới dạng: 'a.b.c.d', trong đó a, b, c, d là các số hệ 10 biểu diễn giá trị các số nhị phân 8 bits. Địa chỉ này được phân một cách logic thành 2 trường: <địa chỉ mạng><địa chỉ máy> (hay <network address><host address>). Phần địa

chỉ mạng nằm ở các bit cao, còn phần địa chỉ máy nằm ở các bit thấp, và độ dài của các trường này phụ thuộc vào lớp địa chỉ mạng được sử dụng.

Có 3 lớp địa chỉ và một vùng địa chỉ để dành như sau:

Các bits cao đầu tiên	Miền giá trị của a	Lớp địa chỉ	Phần địa chỉ mạng	Phần địa chỉ máy
0...	1-126	A	a	b.c.d
10...	128-191	B	a.b	c.d
110...	192-223	C	a.b.c	d
111...	224-254	Các địa chỉ để dành	a.b.c	d

Với những máy thuộc mạng con (subnet) mà mạng con này nằm trong các mạng lớn hơn, thì phần địa chỉ mạng nằm trong địa chỉ IP sẽ được phân thành các phần nhỏ hơn để lưu địa chỉ của các mạng con ấy. Ví dụ 2 trường hợp sau:

Địa chỉ IP	Địa chỉ mạng	Địa chỉ mạng con	Địa chỉ máy	Ý nghĩa
128.17.75.20	128.17		75.20	Chỉ tới máy số 75.20 của mạng 128.17
128.17.75.20	128.17	75	20	Chỉ tới máy số 20 nằm trong mạng con số 75 của mạng 128.17

Một số giá trị đặc biệt (của byte cao):

- 0: địa chỉ của những máy lúc khởi động khi chúng chưa biết IP riêng của nó.
- 127: là địa chỉ 'loopback' được sử dụng khi một máy muốn tạo liên kết TCP/IP tới chính nó.
- 224-254: địa chỉ để dành.
- 255: địa chỉ quảng bá (broadcast).

Mặt nạ của mạng con (subnet mask):

Các bit của mặt nạ có giá trị 1 sẽ chỉ ra rằng các bit ở vị trí tương ứng trong địa chỉ IP sẽ thuộc phần địa chỉ mạng. Còn các bit của mặt nạ có giá trị 0 sẽ chỉ ra rằng các bit ở vị trí tương ứng trong địa chỉ IP sẽ thuộc phần địa chỉ máy.

Ví dụ: một máy có địa chỉ IP là '128.17.75.20' và mặt nạ mạng là '255.255.255.0', sẽ chỉ ra phần địa chỉ mạng con là '128.17.75' và địa chỉ máy là 20. Do '128.17.75.20' thuộc địa chỉ của lớp B (do byte đầu tiên có giá trị 128, mà $128 \leq 128 \leq 191$) nên phần địa chỉ mạng là '128.17'. Do vậy, trong

trường hợp này, địa chỉ IP '128.17.75.20' với mặt nạ mạng là '255.255.255.0', sẽ tham chiếu tới máy số 20 nằm trong mạng con 75 của mạng 128.17.

Người quản trị mạng sẽ tự chọn mặt nạ mạng và cung cấp thông tin này cho các máy con của nó.

Địa chỉ 'loopback' (địa chỉ IP tương ứng: 127.0.0.1) luôn có mặt nạ mạng là 255.0.0.0.

Địa chỉ mạng con (subnetword address):

Địa chỉ này chính là phần địa chỉ mạng con của địa chỉ IP và được xác định bởi mặt nạ mạng.

Ví dụ: máy có địa chỉ IP là '128.17.75.20' và mặt nạ mạng là '255.255.255.0', sẽ cho địa chỉ mạng con là '128.17.75.0'.

Hệ thống 'loopback-only' sẽ không có địa chỉ mạng con.

Địa chỉ quảng bá của mạng (broadcast address):

Địa chỉ này được sử dụng khi muốn chuyển một gói tin tới mọi máy tính trong mạng con.

Ví dụ: một mạng con có địa chỉ là '128.17.0.0' sẽ có địa chỉ quảng bá là '128.17.255.255', tương tự, một mạng con có địa chỉ là '128.17.75.0' sẽ có địa chỉ quảng bá là '128.17.75.255'.

Một số hệ thống sử dụng địa chỉ mạng (địa chỉ mạng con) như là địa chỉ quảng bá. Do vậy cần liên hệ với người quản trị mạng để biết rõ về vấn đề này.

Hệ thống 'loopback-only' sẽ không có địa chỉ quảng bá.

Địa chỉ IP của "gateway":

Đây chính là địa chỉ của những máy tính được cung cấp các phần mềm, các công cụ kiểm soát thích hợp nhằm cho phép các mạng giao tiếp được với nhau (dù chúng sử dụng các giao thức khác nhau). Các thiết bị thực hiện việc kết nối các máy trong mạng của bạn với các mạng khác (kết nối với thế giới bên ngoài) có thể là các: 'router' (bộ chọn đường), 'gateway' (cửa khẩu). Và người quản trị mạng sẽ cung cấp cho bạn địa chỉ IP của các thiết bị kết nối này.

Địa chỉ này không có ở các hệ thống 'loopback-only' hoặc các mạng đơn lẻ.

Địa chỉ IP của "name-server":

Đây chính là địa chỉ IP của máy tính đảm nhận việc chuyển đổi từ 'hostname' thành địa chỉ IP tương ứng. Địa chỉ này sẽ được cung cấp bởi người quản trị hệ thống.

Các hệ thống 'loopback-only' sẽ không có địa chỉ IP của 'name-server'.

5.2.3 Tên máy (hostname)

Trong giao tiếp trên mạng, ta có thể dùng 'hostname' hoặc địa chỉ IP tương ứng. Ví dụ:

Tên máy (hostname)	Địa chỉ IP
Eggplant	128.17.75.20
Apricot	128.17.75.12
Papaya	128.17.112.3, 128.17.75.98
Pear	128.17.112.21
Pineapple	128.17.112.40, 128.17.30.1

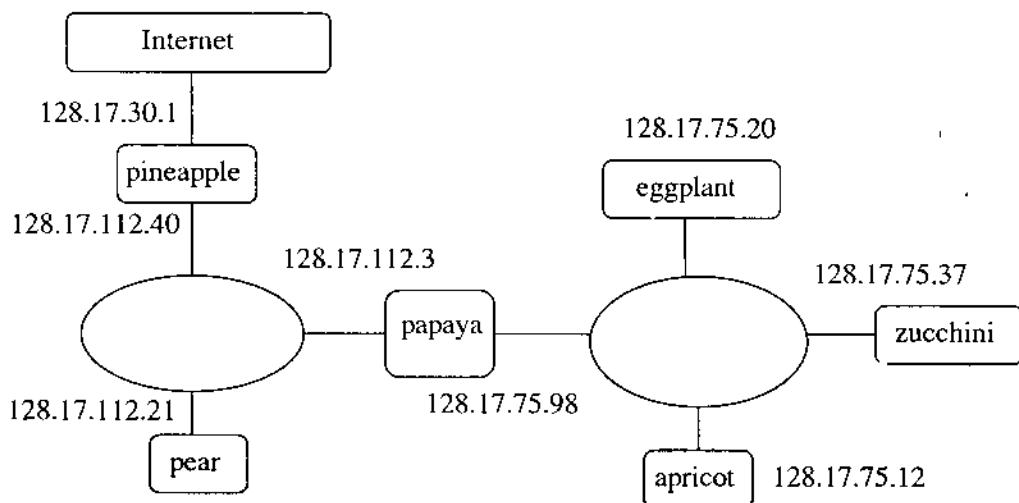
Mỗi 'hostname' có thể có một hoặc nhiều địa chỉ IP, trong ví dụ trên, 'Pineapple' có 2 địa chỉ IP, một là 128.17.112.40 trên mạng con 128.17.112, còn một là 128.17.30.1 trên mạng con 128.17.30.

Việc chuyển từ 'hostname' sang địa chỉ IP tương ứng được hệ thống thực hiện nhờ file '/etc/hosts' hoặc nhờ DNS (Domain Name Service-DNS).

5.2.4 Bảng chọn đường

Để thực hiện việc truyền các gói tin tới đúng đích, hệ thống dựa vào phần địa chỉ mạng (nằm trong địa chỉ IP) để xây dựng các bảng chọn đường (routing table) riêng cho mạng. Bảng chọn đường này sẽ chứa danh sách các mạng và các 'gateway' (cửa khẩu) tương ứng của chúng.

Ví dụ ta có sơ đồ sau:



Trong trường hợp này, bảng chọn đường của máy 'papaya' như sau:

Destination	Gateway	Genmask	Flags	MSS	window	irtt	Iface
128.17.75.0	128.17.75.98	225.225.225.0	UN	1500	0	0	Eth0
128.17.112.0	128.17.112.3	225.225.225.0	UN	1500	0	0	Eth1
default	128.17.112.40	0.0.0.0	UGN	1500	0	0	Eth1
127.0.0.1	127.0.0.1	225.0.0.0	UH	3584	0	0	LO
128.17.75.98	127.0.0.1	225.225.225.0	UH	3584	0	0	LO

Khi nhận một gói tin, hệ thống sẽ đối sánh địa chỉ đích của gói tin với các địa chỉ IP đích (đã được lưu trong bảng chọn đường) để tìm địa chỉ đích thích hợp và địa chỉ của gateway tương ứng để chuyển gói tin tới. Quá trình này sẽ được tiếp tục cho đến khi gói tin được chuyển tới đúng đích.

5.2.5 Một số lệnh cơ bản: *rcp*, *rlogin*, *telnet*, ...

- *rcp* (remote file copy): cho phép copie các file giữa các máy làm việc dưới hệ điều hành Unix
- *rlogin*: cho phép người sử dụng ở các 'terminal' khác nhau 'connect' với nhau.

Ví dụ:

```
[huynq@may9 huynq]$
[huynq@may9 huynq]$ rlogin may8
[huynq@may8 huynq]$
```

- *telnet*: cho phép người sử dụng 'connect' tới một hệ thống khác (đây cũng là một phiên bản phổ biến hơn của công lệnh *rlogin*).

Ví dụ:

```
[root@may11 /mnt]# telnet may9
Trying 192.168.50.9...
Connected to may9.
Escape character is '^]'.

Red Hat Linux release 6.1 (Cartman)
Kernel 2.2.12-20 on an i686
```

```
login: huynq
Password:
[huynq@may9 huynq]$
```

- *talk*: cho phép nói chuyện trực tiếp với một 'user' đang làm việc trên một máy khác.

Ví dụ: từ 'may11', muốn nói chuyện với 'nhson' đang làm việc trên 'may14', hãy đưa lệnh:

```
talk nhson@may14
```

Thông tin hiện trên 'terminal' đang được kích hoạt của 'may14':

```
Message from Talk_Daemon@may14 at 10:03 ...
talk: connection requested by dinhnt@may11.
talk: respond with: talk dinhnt@may11
```

Để đồng ý thiết lập liên kết, 'nhson@may14' sẽ trả lời:

```
talk dinhnt@may11
[Connection established]
...
```

- *finger*: cho phép nhận các thông tin về các 'user' của một máy tính khác (ở xa).

Ví dụ:

```
[huynq@may8 huynq]$ finger
Login      Name      Tty      Idle      Login Time Office
Office Phone
huynq              /0              Mar 25 15:15 (may9)
huynq              /2              Mar 25 15:22 (may11)
linhtd           /1             1 Mar 25 15:21 (may11)
[huynq@may8 huynq]$
```

- *rsh*: cho phép thực hiện một công lệnh shell trên một máy tính khác ở xa.

Ví dụ: dùng 'rsh' để 'connect' từ 'máy8' tới 'máy3':

```
[huynq@may8 /etc]$ rsh may3
Last login: Sat Mar 25 07:41:54 from may11
[huynq@may3 huynq]$
```

Ví dụ: từ 'máy3', dùng 'rsh' để liệt kê mọi tên tất cả các file ở thư mục gốc trên 'máy8', kết quả được hiện thị tại terminal đang làm việc ở 'máy3', sau khi công lệnh được thực hiện xong, liên kết được thiết lập tới 'máy8' sẽ bị đóng:

```
[huynq@may3 huynq]$ rsh may8 | ls -a
.          .bash_history    .enlightenment      .kde
.xpvm_hosts
..          .bash_logout      .gnome                .kderc Desktop
.ICEauthority .bash_profile .gnome-desktop .mc nsmail
.Xauthority .bashrc      .gnome-help-browser .netscape pvm3
.Xdefaults   .cedit        .gnome_private       .screenrc
rlogin: connection closed.
[huynq@may3 huynq]$
```

- *ftp*: cho phép copic các file giữa các hệ thống (đây là một phiên bản phổ biến hơn của công lệnh rcp).
- *ping*: là lệnh đơn giản nhất cho phép kiểm tra các liên kết trong mạng.

Ví dụ:

```
[root@may11 /mnt]# ping may9
PING may9 (192.168.50.9) from 192.168.50.11 : 56(84) bytes
of data.
64 bytes from 192.168.50.9: icmp_seq=0 ttl=255 time=0.2 ms
```

```
[huynq@may9 huynq]$ ping may8
```

```

PING may8 (192.168.50.8) from 192.168.50.9 : 56(84) bytes
of data.
64 bytes from 192.168.50.8: icmp_seq=0 ttl=255 time=0.3 ms
64 bytes from 192.168.50.8: icmp_seq=1 ttl=255 time=0.1 ms
64 bytes from 192.168.50.8: icmp_seq=2 ttl=255 time=0.1 ms
64 bytes from 192.168.50.8: icmp_seq=3 ttl=255 time=0.1 ms
64 bytes from 192.168.50.8: icmp_seq=4 ttl=255 time=0.1 ms
64 bytes from 192.168.50.8: icmp_seq=5 ttl=255 time=0.1 ms
64 bytes from 192.168.50.8: icmp_seq=6 ttl=255 time=0.1 ms
64 bytes from 192.168.50.8: icmp_seq=7 ttl=255 time=0.1 ms

--- may8 ping statistics ---
8 packets transmitted, 8 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.3 ms
[huyngq@may9 huyngq]$

```

- *who*: được dùng để liệt kê các 'user' đang 'login' vào máy hiện tại:

Ví dụ:

```

[nhson@may14 msdos]$ who
nhson      tty1      Apr  1 09:39
huyngq     tty2      Apr  1 09:56
nhson      pts/0      Apr  1 09:47
nhson      pts/1      Apr  1 09:50
[nhson@may14 msdos]$

```

- *ifconfig*: lệnh cho phép cấu hình các giao thức trên mạng.
- *route, routed*: lệnh cho phép cấu hình việc chọn đường cho các gói tin trên mạng.
- *netstat*: lệnh này sẽ hiển thị mọi liên kết đang hoạt động trên máy hiện tại.

Ví dụ:

```
[huyngq@may8 huyngq]$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address      Foreign Address
State
tcp        0      0 may8:login         may11:1023
ESTABLISHED tcp        0      0 may8:telnet        may11:1079
ESTABLISHED tcp        0    124 may8:login         may9:1023
ESTABLISHED Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags      Type       State      I-Node Path
unix    0      [ ]       STREAM     CONNECTED  302 @000000017
unix    5      [ ]       DGRAM              412 /dev/log
unix    0      [ ]       STREAM     CONNECTED  107 @00000000f
unix    0      [ ]       DGRAM              1090
unix    0      [ ]       DGRAM              1022
unix    0      [ ]       DGRAM              979
unix    0      [ ]       DGRAM              629
unix    0      [ ]       DGRAM              427
[huyngq@may8 huyngq]$
```

- *netstat -rn*: cho phép hiển thị bảng chọn đường.

Để biết cách sử dụng chi tiết của các công lệnh này, bạn có thể xem phần trợ giúp hướng dẫn cách sử dụng của nó qua lệnh *man*.

Ví dụ: *man route* sẽ hiển thị trong trợ giúp của lệnh *route*.

5.2.6 NFS và NIS

NFS (Network File System -NFS) cho phép làm việc (truy cập) tới một hệ thống file ở xa như làm việc đối với hệ thống file được lưu trữ trực tiếp trên máy.

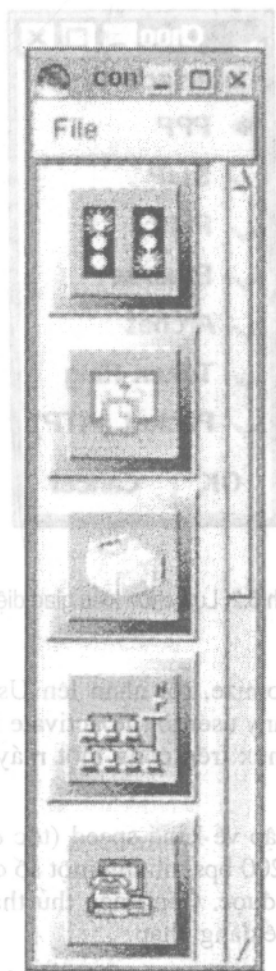
NIS (Network Information Service -NIS) cho phép quản trị cấu hình của toàn hệ thống và tập trung hóa một số công việc về quản trị trên toàn mạng.

NFS và NIS đều hoạt động dưới mạng TCP/IP, trên hệ điều hành UNIX cũng như không UNIX.

5.3 KẾT NỐI INTERNET

Trước hết, để kết nối được với Internet, bạn phải qua một nhà cung cấp dịch vụ Internet (ISP) và bạn phải liên hệ với ISP của mình để biết được số điện thoại truy cập cũng như các thiết lập DNS và gateway. Trong trường hợp bạn đã có một mạng cục bộ kết nối với Internet, bạn chỉ việc thiết lập gateway và proxy (nếu có) cho máy con của bạn là được.

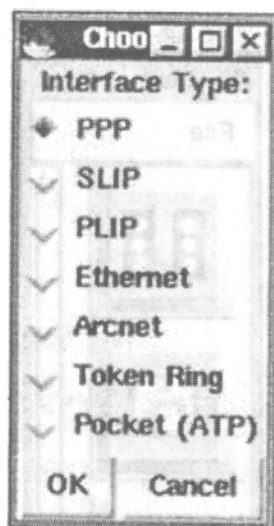
Trong X Windows Manager, bạn chạy *Linuxconf*, hiển thị nó trên màn hình. Nhấn vào biểu tượng modem chỉ ra cổng (port - device) có gắn modem của bạn (hình 5.1).



Hình 5.1. Lựa chọn cổng cho modem.

Bước tiếp theo là chạy công cụ cấu hình mạng bằng cách nhấn vào biểu tượng mạng. Nhập số IP của DNS mà bạn đã được ISP cung cấp vào hộp Nameserver. Đồng thời, bạn có thể tùy ý nhập hostname trong Hostname và domain trong Domain - chúng cũng giống như của ISP. Ví dụ người dùng "netvigator" có thể nhập "netvigator.com" ở đây.

Nhấn vào nút Interfaces, sau đó nhấn Add. Chọn 'PPP' làm Interface Type (Hình 5.2). Sau đó nhập số điện thoại của ISP, tên login PPP và password (nếu bạn đăng nhập không được, quay lại và nhấn vào nút PAP để sử dụng Password Authentication Protocol).



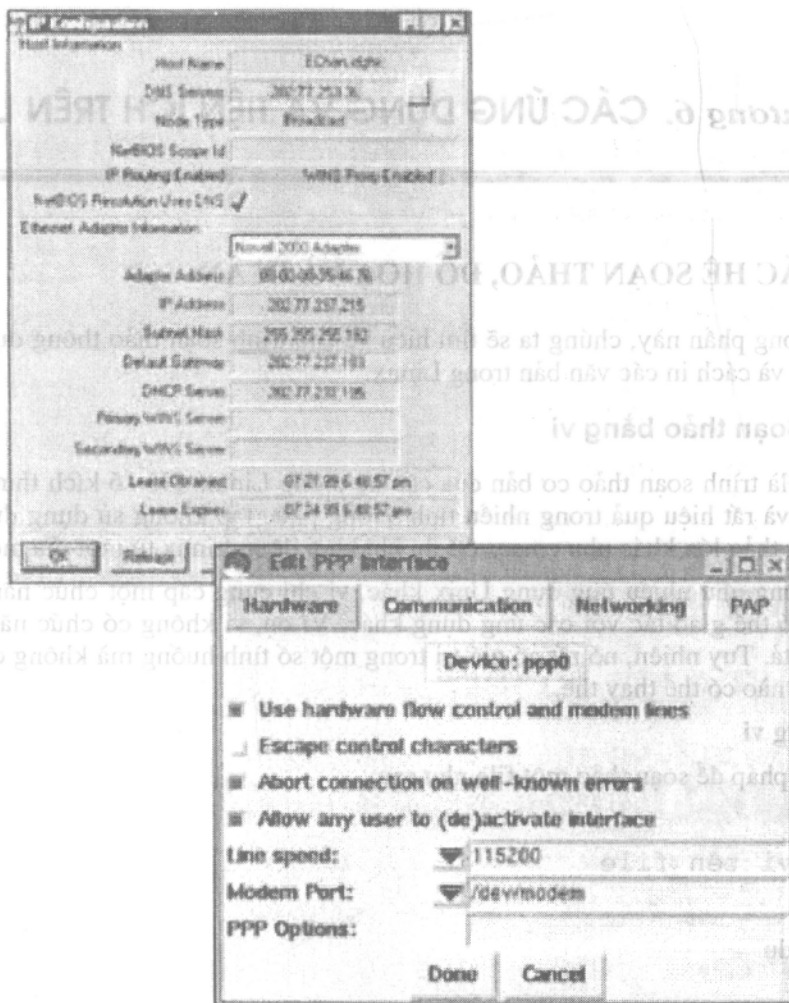
Hình 5.2. Lựa chọn kiểu giao diện.

Nhấn tiếp vào nút Customize, rồi nhấn lên Use hardware flow control and modem lines, và chọn 'Allow any user to (de)activate interface'. Điều này cho phép những người khác sử dụng Linux trên cùng một máy tính để kết nối vào Internet bằng thiết lập dial-up của bạn.

Hãy cẩn thận với thiết lập về Line speed (tốc độ truyền). Mặc dù phần lớn modem làm việc ở tốc độ 115.200 bps, nhưng một số chỉ ở 57.600 bps. Giảm giá trị này xuống nếu kết nối không được. Cũng nên thử thay đổi chuỗi ký tự nằm dưới thiết lập PAP nếu bạn không thể đăng nhập.

Nhấn vào nút Networking. Sau khi chắc chắn là kết nối làm việc tốt, bạn có thể chọn Restart PPP khi không kết nối được: hệ thống tự động thiết lập lại kết nối

khi đường truyền bị ngắt. Nếu có các máy tính khác kết nối vào máy bạn, bạn cũng nên chọn "Set default route when making connection" (bảng chọn đường mặc định).



Hình 5.3. Đặt các thông số.

Khi tất cả đã xong, nhấn vào Save để lưu lại các thiết lập. Mỗi khi kết nối vào Internet, bạn gọi đến Network Configuration Setting (hay gõ "netcfg") trong linuxconf, và nhấn vào ppp0, rồi chọn Activate.

Chương 6. CÁC ỨNG DỤNG VÀ TIỆN ÍCH TRÊN LINUX

6.1 CÁC HỆ SOẠN THẢO, ĐỒ HOẠ VÀ IN ẤN

Trong phần này, chúng ta sẽ tìm hiểu về các trình soạn thảo thông dụng như vi, emacs và cách in các văn bản trong Linux.

6.1.1 Soạn thảo bằng vi

Vi là trình soạn thảo cơ bản của các hệ thống Linux. Nó có kích thước nhỏ, đơn giản và rất hiệu quả trong nhiều tình huống phức tạp không sử dụng được các trình soạn thảo lớn khác như emacs (ví dụ khi khởi động Linux từ một đĩa mềm).

Giống như nhiều ứng dụng Unix khác, vi chỉ cung cấp một chức năng nhất định và có thể giao tác với các ứng dụng khác. Ví dụ, vi không có chức năng soát lỗi chính tả. Tuy nhiên, nó rất có giá trị trong một số tình huống mà không có trình soạn thảo nào có thể thay thế.

Khởi động vi

Cú pháp để soạn thảo một file như sau:

```
$ vi tên_file
```

Ví dụ

```
$ vi test
```

Sẽ cho phép soạn thảo một file có tên là test.

Chèn văn bản

Khi sử dụng vi, bạn có thể đặt một trong ba chế độ thao tác: chế độ câu lệnh, chế độ soạn thảo và chế độ mở rộng.

Sau khi khởi động vi, bạn ở trong chế độ câu lệnh. Trong chế độ này, bạn có thể sử dụng một số câu lệnh (thường là các chữ đơn lẻ) để sửa đổi văn bản. Văn bản thường được chèn và sửa đổi trong chế độ soạn thảo. Để bắt đầu chèn văn bản, bấm phím i (chuyển sang chế độ soạn thảo), sau đó bạn có thể bắt đầu soạn thảo văn bản. Trong khi soạn thảo văn bản, bạn có thể dùng Enter để xuống dòng và Backspace để sửa lỗi. Để kết thúc chế độ soạn thảo và quay trở lại chế độ câu lệnh, bấm phím Esc.

Trong chế độ câu lệnh ...

Bạn có thể dùng các phím h, j, k và l để dịch chuyển con trỏ sang trái, sang phải, lên trên và xuống dưới.

Một cách khác để chèn văn bản mà không cần dùng lệnh i (gõ phím i) là bạn có thể dùng lệnh a (bấm phím a) để chèn văn bản sau vị trí hiện thời của con trỏ. Ví dụ, dùng phím mũi tên để dịch chuyển con trỏ đến giữa chữ học và khoa:

```
Truong dai hoc _ khoa Ha Noi
```

Bấm phím a, gõ Bach và bấm ESC để quay lại chế độ câu lệnh

```
Truong dai hoc Bach khoa Ha Noi
```

Để gõ một dòng văn bản mới dưới dòng văn bản hiện thời, dùng lệnh o (gõ phím o)

```
Truong dai hoc Bach khoa Ha Noi
```

```
Khoa Cong nghe thong tin_
```

Bạn chỉ có thể sử dụng các lệnh i, a, o trong chế độ câu lệnh. Nếu bạn không biết đang ở chế độ nào, hãy gõ phím ESC, bạn sẽ quay về chế độ câu lệnh nếu như bạn đang ở chế độ soạn thảo hoặc sẽ có một tiếng bíp nếu như bạn đang ở chế độ câu lệnh.

Xoá đoạn văn bản và loại bỏ những thay đổi với văn bản

Để xoá các ký tự phía trước con trỏ, dùng lệnh x. Ví dụ nếu bạn gõ phím x 4 lần, bạn sẽ có văn bản như sau

```
Truong dai hoc Bach khoa Ha Noi
```

Khoa Công nghệ thông tin

Dùng lệnh x (gõ phím d 2 lần) để xoá toàn bộ một dòng. Bạn đang ở dòng thứ 2 , gõ phím d hai lần, bạn sẽ có

```
_Truong dai hoc Bach khoa Ha Noi
```

Dùng lệnh p để chèn dòng văn bản đã xoá từ bộ đệm vào sau dòng văn bản hiện thời.

Lệnh u cho phép khôi phục lại văn bản trước lần thay đổi sau cùng. Trong ví dụ trên, bấm u sau khi bấm x thì kết quả tương đương với bấm p.

Dùng dw để xoá toàn bộ từ tại vị trí con trỏ. Ví dụ, chuyển con trỏ đến từ công và gõ dw, bạn sẽ có:

```
Truong dai hoc Bach khoa Ha Noi
Khoa _ nghe thong tin
```

Con trỏ sẽ định vị tại từ tiếp theo

Các lệnh dịch chuyển con trỏ

Lệnh w chuyển con trỏ tới ký tự đầu tiên của từ tiếp theo. Bấm b chuyển con trỏ tới ký tự đầu tiên của chữ hiện thời. Lệnh 0 (số 0) chuyển con trỏ tới đầu dòng, lệnh \$ chuyển con trỏ tới cuối dòng.

Bấm Ctrl —B chuyển nó xuống dưới một trang màn hình.

Dùng lệnh G để chuyển con trỏ tới cuối file, lệnh 10G chuyển con trỏ tới dòng thứ 10 của file, còn để chuyển con trỏ tới đầu file, gõ 1G.

Để dịch chuyển con trỏ đến ký tự theo mẫu nào đó, có thể dùng ký tự /. Ví dụ, chuyển con trỏ đến đầu file trong ví dụ trên và gõ /Cộng hoà Nhân dân Trung Hoa, con trỏ sẽ được chuyển đến từ thông. Dùng ? thay cho / cho phép dịch con trỏ theo hướng ngược lại.

Để tìm một chữ hoa tiếp theo, bạn có thể sử dụng /[A-Z].

Ghi file và thoát khỏi vi.

Gõ : w cho phép ghi file

Gõ : wq để ghi file và thoát khỏi vi (có thể dùng lệnh zz cũng cho kết quả tương tự)

Gõ : q! để thoát khỏi vi mà không ghi lại các thay đổi đã thực

Các lệnh dịch chuyển con trỏ

Lệnh `w` chuyển con trỏ tới ký tự đầu tiên của từ tiếp theo. Lệnh `b` chuyển con trỏ tới ký tự đầu tiên của chữ hiện thời. Lệnh `0` (số 0) chuyển con trỏ tới đầu dòng, lệnh `$` chuyển con trỏ tới cuối dòng.

Bấm `Ctrl —B` chuyển nó xuống dưới một trang màn hình.

Dùng lệnh `G` để chuyển con trỏ tới cuối file, lệnh `10G` chuyển con trỏ tới dòng thứ 10 của file, còn để chuyển con trỏ tới đầu file, gõ `1G`.

Để dịch chuyển con trỏ đến ký tự theo mẫu nào đó, có thể dùng ký tự `/`. Ví dụ, chuyển con trỏ đến đầu file trong ví dụ trên và gõ `/Cộng hoà Nhân dân Trung Hoa`, con trỏ sẽ được chuyển đến từ thông. Dùng `?` thay cho `/` cho phép dịch con trỏ theo hướng ngược lại.

Để tìm một chữ hoa tiếp theo, bạn có thể sử dụng `/[A-Z]`.

Ghi file và thoát khỏi vi

Gõ : `w` cho phép ghi file

Gõ : `wq` để ghi file và thoát khỏi vi (có thể dùng lệnh `zz` cũng cho kết quả tương tự)

Gõ : `q!` để thoát khỏi vi mà không ghi lại các thay đổi đã thực hiện đối với file.

Soạn thảo một file khác

Dùng lệnh `:e` để soạn thảo một file khác. Ví dụ, để dùng việc soạn thảo file `file test` và soạn thảo file `new`, ta gõ:

```
Truong dai hoc Bach Khoa Ha Noi
Khoa Cong nghe thong tin
...
...
:e new-
```

Bạn sẽ nhận được một thông báo:

```
No write since last change (edit ! overrites)
```

Bạn có thể dùng `:w` để ghi lại file `test` rồi dùng lệnh `:e`, hoặc dùng lệnh `:e! new` nếu muốn soạn thảo file `new` mà không cần ghi lại các thay đổi đối với file `test`.

Chèn nội dung file khác

Dùng lệnh `:r`, bạn có thể chèn nội dung của file khác vào file đang soạn thảo.

Ví dụ:

```
: r old.txt
```

cho phép chèn nội dung của file old.txt vào sau dòng đang soạn thảo.

Chạy các lệnh của Shell

Lệnh `!!` Cho phép thực hiện các lệnh của Shell trong khi đang chạy vi. Ví dụ:

```
:! ls -F
```

sẽ thực hiện lệnh `ls` và hiển thị kết quả ra màn hình.

Tìm kiếm và thay thế

Lệnh:

```
: [x,y] s | pattern | replacement | flags
```

sẽ tìm tất cả các pattern và thay thế bằng replacement.

Ví dụ:

```
: 1,10 s | weeble | wobble
```

sẽ thay thế 'weeble' bởi 'wobble' trong đoạn văn bản từ dòng 1 đến dòng 10.

6.1.2 Trình soạn thảo Emacs

Ngoài vi, Emacs là trình soạn thảo thường được sử dụng trong Linux. Hiện nay, có hai kiểu Emacs. GNU Emacs là phiên bản gốc, tuy vẫn được phát triển nhưng rất chậm. Xemacs lớn hơn và thân thiện với người sử dụng hơn.

Khởi động Emacs

```
$ emacs option (với GNU Emacs)
```

hoặc

```
$ xemacs option (với XEmacs)
```

thông thường, ta không cần phải đưa option vào.

Trong Emacs, C-x có nghĩa là Ctrl-X và M-p tương đương với Alt-P, C-M-p có nghĩa là Ctrl-Alt-p.

Các lệnh soạn thảo đơn giản

- C-p: dịch con trỏ đến dòng trước.
- C-n: dịch con trỏ đến dòng tiếp theo.
- C-f: dịch con trỏ đến kí tự trước.
- C-b: dịch con trỏ đến kí tự sau.
- C-a: dịch con trỏ đến đầu dòng.
- C-e: dịch con trỏ đến cuối dòng.
- C-v: dịch con trỏ xuống dưới một trang.
- M-v: dịch con trỏ lên trên một trang.
- Để thoát khỏi Emacs, sử dụng C-x C-c.

Xem trợ giúp

Bấm C-h để xem trợ giúp.

Xoá, sao chép, và dịch chuyển khối văn bản

Dịch chuyển con trỏ tới đầu khối văn bản cần đánh dấu, bấm C-@ để bắt đầu đánh dấu, sau đó dịch chuyển con trỏ tới cuối khối văn bản cần đánh dấu, bấm C-@.

Để xoá khối văn bản đã đánh dấu, bấm C-w.

Để sao chép khối văn bản đã đánh dấu, dịch chuyển con trỏ tới vị trí cần sao chép, và bấm C-y.

Tìm kiếm và thay thế

Bấm C-s để tìm một chuỗi kí tự trong Emacs. Nếu Emacs không tìm thấy, sẽ có một tiếng bip, nếu Emacs tìm thấy nhưng bạn muốn tìm từ khác, bấm lại C-s.

Bạn có thể tìm theo chiều từ cuối văn bản đến đầu văn bản bằng cách bấm C-r.

Thực hiện các công lệnh và lập trình trong Emacs

Khi đang chạy Emacs, bạn có thể thực hiện các câu lệnh hoặc chạy các ứng dụng khác.

Để gửi mail trong khi đang chạy Emacs, bấm C-x m, sau đó soạn thảo nội dung thư điện tử và bấm C-c C-s để gửi nó đi:

To:
 Subject:
 -- test follows this line--
 trung dai hoc Bach Khoa Ha Noi

RMAIL là giao diện của Emacs cho phép đọc thư điện tử. Sử dụng lệnh `M-x rmail` để khởi động RMAIL. Bấm `C-h` để xem trợ giúp chi tiết.

Ngoài ra Emacs có rất nhiều chế độ soạn thảo, các kiểu file, ví dụ: các file nguồn của C, Lispp, Text, ...

6.1.3 Sử dụng ImageMagick

Đây là một chương trình cho phép chuyển đổi và thao tác trên các file ảnh thông qua câu lệnh, điều này rất tiện lợi khi chúng ta cần phải xử lý nhiều file ảnh một lúc.

- *display*: là một chức năng của ImageMagick cho phép bạn kiểm tra các kết quả của một chức năng nào đó trước khi bạn thật sự thực hiện chức năng đó với hàng loạt file ảnh.

- *import*: cho phép thu thập thông tin một cửa sổ hoặc toàn bộ màn hình. Ví dụ nếu bạn muốn ghi lại màn hình vào file `myscreen.tiff`, bạn gõ lệnh:

```
tigger$ import -window root myscreen.xpm
```

Nếu bạn chỉ muốn lấy một cửa sổ:

```
tigger$ import myscreen.xpm
```

- *montagne*: cho phép tạo một ảnh lớn bằng cách ghép một số ảnh nhỏ với nhau. Ví dụ, ghép tất cả các ảnh JPEG trong thư mục hiện thời để tạo thành ảnh `all.jpg` như sau:

```
tigger$ montagne *.jpg all.jpg
```

Theo mặc định, dưới mỗi ảnh đều có nhãn ghi tên của nó. Bạn có thể bỏ đi bằng cách:

```
tigger$ montagne +frame *.jpg all.jpg
```

- *convert*: cho phép chuyển ảnh từ khuôn dạng này sang khuôn dạng khác. Ví dụ để chuyển các biểu tượng dạng BMP trong Windows thành dạng XPM:

```
for i in *.bmp
do
    convert $i xpm:'basename $i.bmp'.xpm
done
```

- *mogrify*: tương tự như convert nhưng nó ghi đè ảnh mới lên ảnh gốc.
- *identify*: hiển thị các thông số của ảnh.

Ví dụ:

```
tigger$ identify tux.gif
tux.gif      257x303+0+0 DirectClass 1096b GIF is
```

- *Combine*: tổ hợp nhiều ảnh thành một ảnh. Bạn có thể đặt một biểu tượng lên trên một ảnh lớn hơn.

- *xtp*: giống như ứng dụng ftp, xtp cho phép tải các file ảnh từ một trang Web ở xa hoặc cập nhật lên một trang Web ở xa.

Ngoài ImageMagick, Linux còn hỗ trợ một số chương trình xử lý ảnh khác như GIMP (GNU Image Manipulation Program), POVRAY, ...

6.1.4 In ấn

Lệnh `lpr` cho phép in một văn bản trong Linux. Ví dụ để in file `vd.c`, bạn gõ lệnh:

```
lpr vd.c
```

`lpr` bắt đầu tiến trình in bằng cách lưu trữ dữ liệu vào một thư mục được gọi là print spool. Mỗi máy in trong hệ thống thường có ít nhất một print spool. Theo mặc định, `lpr` sẽ tìm máy in có tên `lp`. Nếu bạn cần chỉ rõ máy in có tên khác, bạn phải có thêm lựa chọn `-P`. Ví dụ, để in văn bản bằng máy in có tên là `nene`, bạn gõ lệnh:

```
lpr -Pnene vd.c
```

6.2 TIỆN ÍCH MC

`mc` (midnight commander) là tiện ích đi kèm theo Linux. Ý tưởng của `mc` được xuất phát từ bộ tiện ích `nc` (Norton Commander) trên hệ điều hành DOS. Nếu bạn đã quen sử dụng `nc` trên DOS thì việc chuyển sang dùng `mc` không có gì khác

so với nc cả. Nếu bạn chưa sử dụng qua nc thì cũng không sao, mc là một tiện ích giúp bạn thực hiện các lệnh của shell một cách nhanh chóng.

Khi khởi động mc, trên màn hình hiện ra hai cửa sổ được xếp cạnh nhau, mỗi cửa sổ là nội dung của một thư mục, như vậy, cùng lúc bạn có thể làm việc trên hai thư mục. Bạn có thể thực hiện các thao tác về tệp trên mỗi cửa sổ hoặc copy, chuyển đổi tệp giữa hai cửa sổ này một cách hết sức dễ dàng. Sau đây là một số lệnh cơ bản trong mc :

- F1 : hiện ra giúp đỡ
- F2 : menu của người dùng
- F3 : xem file hay thư mục
- F4 : soạn thảo tệp
- F5 : sao chép tệp
- F6 : đổi tên/di chuyển tệp
- F7 : tạo thư mục con
- F8 : xoá tệp hoặc thư mục
- F9 : menu lựa chọn
- F10 : thoát khỏi chương trình

6.3 INTERNET VÀ WEB/MAIL

6.3.1 Giới thiệu về Web

Web là khái niệm mà người dùng Internet quan tâm nhiều nhất hiện nay. Người ta viết tắt là WWW hay gọi ngắn gọn là Web. Web là một công cụ, hay đúng hơn là dịch vụ của Internet. Khác với các dịch vụ trước đây của Net, Web chứa thông tin bao gồm văn bản, hình ảnh, âm thanh và thậm chí cả video được kết hợp với nhau. Web cho phép bạn có thể chui vào mọi ngõ ngách trên Net, là những điểm chứa CSDL gọi là Web site. Nhờ có Web, nên dù không phải là một chuyên gia, bạn vẫn có thể sử dụng Internet. Phần mềm sử dụng để định hướng Web gọi là trình duyệt (browser). Hiện nay, trình duyệt thông dụng nhất là Navigator của Netscape, tiếp đó là Internet Explorer của Microsoft (kèm theo hệ điều hành Windows 95).

6.3.2 Thư điện tử (E-mail)

Đây là dịch vụ của Internet được sử dụng nhiều nhất. Bạn có thể trao đổi thư (e-mail) với hàng triệu người khắp thế giới. Người ta sử dụng e-mail vào bất cứ việc gì mà họ có thể làm với giấy hay điện thoại: bàn công việc, tán gẫu, hỏi thăm, tỏ tình... và cả những chuyện phi pháp nữa. Danh sách thư điện tử (mailing list) cho

phép bạn gia nhập vào những cuộc bàn luận theo nhóm người có cùng mối quan tâm và gặp gỡ thông qua mạng. Dịch vụ thư tín (Mail Servers) giúp bạn truy cập những thông tin cần thiết.

6.3.3 Trình duyệt Web netscape

Trình duyệt Netscape Navigator là trình duyệt thông dụng nhất trên Linux. Trình duyệt này được kèm theo trong bộ đĩa cài đặt hệ điều hành Linux. Giao diện đồ họa của chương trình này giúp cho người sử dụng Internet có thể lướt trên biển dữ liệu. Để xem các trang Web, bạn có thể gõ trực tiếp địa chỉ trang Web vào phần Address (địa chỉ) sau đó bạn sử dụng các nút Tiến (Forward) hay Lùi (Backward) để lật qua các trang Web.

Để thiết lập các thông tin về địa chỉ email hay mail server, bạn vào mục Edit\Preferences\Mail&Newgroups.

Để thiết lập các thông tin về proxy, cache, bạn vào mục menu Edit\Preferences\Advanced.

Phụ lục 1. HƯỚNG DẪN CÀI ĐẶT REDHAT 6.1

P1. MỘT SỐ LƯU Ý TRONG CÀI ĐẶT REDHAT LINUX 6.1

P1.1. Các đặc điểm mới

Red Hat Linux 6.1 cung cấp ‘Installation-Related Enhancements’ (các tăng cường cài đặt các phần có liên quan) làm cho việc cài đặt Red Hat Linux được dễ dàng hơn.

Sau đây là một số thay đổi đó:

- GUI Installation Mode
- Customizable Workstation-Class Installations
- More Flexible Workstation- and Server-Class Installations
- New Scriptable Kickstart Mode
- User Account Creation
- Software RAID Support
- Up Front Configuration

1. GUI Installation Mode

Việc sử dụng giao diện đồ họa (GUI) để cài đặt cho phép sử dụng chuột trong quá trình cài đặt, các màn hình đồ họa giúp người cài đặt có cách nhìn trực quan hơn, và cũng cung cấp cả các màn hình trợ giúp trong quá trình cài đặt.

2. Customizable Workstation-Class Installations

Với cài đặt trong ‘workstation-class’, bạn có thể chọn hệ thống ‘X Window’ chuẩn và GNOME hoặc KDE thay cho GNOME. Các lựa chọn này bảo đảm các package mong muốn đều được cài đặt và cũng đem lại cho bạn cách cài đặt đơn giản nhất.

3. *Flexible Workstation- and Server-Class Installations* (các cài đặt mềm dẻo trong Workstation-Class và Server-Class)

Cả 2 cách cài đặt trong Workstation-Class và Server-Class đều cho phép khả năng phân hoạch các thiết bị cứng bằng tay. Nếu không vừa ý với cách phân hoạch đã được thực hiện trước đó, bạn có thể phân hoạch lại theo ý muốn trong cách cài đặt theo 'custom-class'.

4. *New Scriptable Kickstart*

Red Hat Linux 6.1 cung cấp nhiều lựa chọn khác nhau cho việc khởi động cài đặt.

5. *User Account Creation*

Trong quá trình cài đặt Red Hat Linux 6.1, bạn có thể tạo 'root password' cho máy cũng như các 'user account' cần thiết.

6. *Software RAID Support*

Trong quá trình cài đặt, bạn có thể chọn cài đặt các thiết bị RAID (RAID devices) cho một hoặc một số thiết bị cứng khi cài đặt các partition.

7. *Up Front Configuration*

Điều này cho phép không tạo bất kỳ thay đổi nào trên hệ thống của bạn trước khi chưa hoàn tất việc cấu hình Red Hat Linux. Chương trình cài đặt sẽ ghi tất cả các 'configuration files' vào bộ nhớ, và chỉ sau khi hoàn thiện việc cấu hình X thì mới tạo các thay đổi thật sự lên hệ thống.

P1.2. Cần một 'Network Boot Disk'?

Bạn sẽ cần một 'network boot disk' riêng nếu thực hiện cài đặt qua FTP, HTTP, hoặc NFS. File 'network boot disk image' là 'bootnet.img' có trong thư mục 'images' trên đĩa CD 'Red Hat Linux/Intel'.

P1.3. Cần một 'PCMCIA Boot Disk'? (đối với các máy 'Intel-based')

Bạn sẽ cần một đĩa khởi động PCMCIA ở một trong các trường hợp sau:

- Cài đặt từ một 'PCMCIA-Connected CD-ROM', tức bạn sẽ cài đặt Red Hat Linux từ một được gắn với một card PCMCIA.

- Sử dụng 'PCMCIA Network Card' (hay 'PCMCIA Network adapter') trong quá trình cài đặt.

Đĩa khởi động PCMCIA cần được tạo trong trường hợp cần thiết. File 'PCMCIA boot disk image' là 'pcmcia.img', có trong thư mục 'images' trên đĩa CD 'Red Hat Linux/Intel'.

P1.4. Tạo đĩa mềm khởi động

Trước tiên, bạn phải có một đĩa mềm trắng 1.44 MB, 3.5-inch, đã format trước, và đưa đĩa này vào ổ mềm, chạy ở chế độ DOS hoặc ở dạng các trình ứng dụng của hầu hết các hệ điều hành Linux.

a) Tạo đĩa mềm khởi động dưới MS-DOS

Đưa đĩa mềm được làm đĩa khởi động vào ổ đĩa mềm (coi là ổ a:), và thực hiện các công lệnh sau (coi ổ CD là ổ d:):

```
C:\> d:
D:\> cd \dosutils
D:\dosutils> rawrite
Enter disk image source file name: ..\images\boot.img
Enter target diskette drive: a:
Please insert a formatted diskette into drive A: and
Press --ENTER-- : Enter
D:\dosutils>
```

b) Tạo đĩa mềm khởi động dưới Linux (giống như hệ O/S)

Trong Linux, ổ mềm được 'mount' (tải) với thiết bị có tên là /dev/fd0. Sau khi tải Red Hat Linux cài đặt bằng lệnh:

```
mount [-t iso9660 /dev/cdrom] /mnt/cdrom
```

bạn hãy chuyển tới thư mục có chứa file ảnh cần thiết (là file có tên là 'boot.img' cho ổ CD, ổ cứng, hoặc 'bootnet.img' nếu muốn cài qua mạng) bằng công lệnh sau:

```
cd /mnt/cdrom/images
```

Tiếp theo, hãy đưa vào lệnh sau:

```
# dd if=boot.img of=/dev/fd0 bs=1440k
```

Nếu muốn tạo thêm một đĩa mềm khác, hãy chạy lại lệnh dd trên với các file image tương ứng.

P1.5. Lưu ý về Kernel Drivers

Trong quá trình cài đặt Red Hat Linux, hạt nhân (kernel) cần được hỗ trợ bởi một số thiết bị như: các thiết bị IDE (bao gồm các thiết bị ATAPI CD-ROM), SCSI adapters, và card mạng; ngoài ra còn có thể có cả: SLIP, CSLIP, PPP, PLIP, FPU emulation, 'console selection', ELF, SysV IPC, 'IP forwarding', 'firewalling', các máy in song song,...

Các hỗ trợ này có thể được cài đặt riêng lẻ theo các modul. Sau khi cài đặt xong, bạn có thể cấu hình lại kernel cho phù hợp với cấu hình phần cứng thực tế của bạn.

P2. BẮT ĐẦU CÀI ĐẶT

Trong phần này, ta sẽ xem xét các nội dung sau:

- Giao diện của chương trình cài đặt với người sử dụng;
- Bắt đầu chương trình cài đặt;
- Lựa chọn phương pháp cài đặt;

P2.1. Giao diện của chương trình cài đặt với người sử dụng

Với giao diện đồ họa, trong quá trình cài đặt, bạn chỉ cần sử dụng chuột để di chuyển trong màn hình, "click" vào các phím (buttons) hoặc đánh vào các dòng text khi cần. Bạn cũng có thể sử dụng phím "Tab" hoặc phím "Enter" để di chuyển giữa các cửa sổ.

- Một số bàn giao tiếp ảo (Virtual Consoles)

Bảng 3-1. Console, phím nóng và nội dung

Console	Phím nóng	Nội dung
1	Ctrl-Alt-F1	Installation dialog (các hộp thoại của chương trình cài đặt)
2	Ctrl-Alt-F2	Shell prompt
3	Ctrl-Alt-F3	Install log (các thông báo từ chương trình cài đặt)
4	Ctrl-Alt-F4	System log (các thông báo từ kernel, ...)
5	Ctrl-Alt-F5	other messages (các thông báo khác)
7	Ctrl-Alt-F7	X graphical display (màn hình đồ họa X)

Chương trình cài đặt cung cấp một số thông báo chuẩn đoán “diagnostic messages” cho người sử dụng, và 5 “virtual consoles” để giúp họ có thể xác định một cách chính xác các vấn đề gặp phải trong quá trình cài đặt Red Hat Linux, qua đó người sử dụng có thể đưa vào các lệnh thích hợp từ “shell prompt”.

Bảng sau đây liệt kê danh sách của 5 “virtual consoles”, phím nóng và nội dung tương ứng của từng “virtual consoles”:

Thông thường, ta ở “virtual console” ngấm định (virtual console #7) khi không gặp phải vấn đề gì trong quá trình cài đặt.

P2.2. Khởi động chương trình cài đặt Red Hat Linux

Việc cài đặt Red Hat Linux chỉ có thể được bắt đầu sau khi bạn đã chắc chắn rằng bạn đã chuẩn bị đầy đủ các tài nguyên cần thiết cho việc cài đặt.

Để khởi động chương trình cài đặt, đầu tiên bạn hãy chèn đĩa khởi động vào ổ đĩa thứ nhất của máy tính rồi khởi động lại, hoặc khởi động thông qua ổ CD-ROM của máy (nếu có). BIOS của máy có thể cần được thay đổi để cho phép bạn khởi động từ ổ đĩa hoặc từ CD-ROM.

Chú ý: để thay đổi BIOS, ngay sau khi khởi động máy và nhìn thấy dòng chữ “press Del to enter Setup”, bạn hãy bấm phím “Del” để vào BIOS và thay đổi lại thứ tự khởi động sao cho máy của bạn có thể được khởi động đầu tiên từ ổ CD-ROM hoặc từ đĩa cứng.

Có 4 phương pháp khởi động sau đây:

- đĩa khởi động cục bộ (‘local boot disk’) -- được sử dụng khi máy của bạn không cho khởi động từ CD-ROM nhưng bạn muốn cài đặt từ ổ CD-ROM cục bộ (‘local CD-ROM’) hoặc từ đĩa cứng.
- đĩa khởi động qua mạng (‘network boot disk’) -- được sử dụng khi khởi động thông qua NFS, FTP và HTTP.
- đĩa khởi động PCMCIA (‘PCMCIA boot disk’) -- được dùng khi bạn cần trợ giúp của PCMCIA (‘PCMCIA support’) trong khi máy của bạn không cho phép khởi động từ ổ CD-ROM, hoặc bạn cần trợ giúp của PCMCIA để cho phép bạn sử dụng CD-ROM trên hệ thống của bạn. Đĩa khởi động này cho phép bạn áp dụng được tất cả các phương pháp cài đặt (CD-ROM, ổ cứng, NFS, FTP, và HTTP).
- đĩa khởi động CD-ROM (‘bootable CD-ROM’) -- được dùng khi máy của bạn cho phép khởi động từ ổ CD-ROM và bạn muốn cài đặt từ ổ CD-ROM cục bộ.

Sau một khoảng thời gian trễ, trên màn hình sẽ hiện lên các lựa chọn khởi động (‘boot options’). Mỗi lựa chọn sẽ có các màn hình trợ giúp đi kèm mà ta có thể truy nhập vào bằng cách bấm các phím tương ứng hiện lên ở dòng cuối của màn hình.

Lưu ý:

- Từ màn hình lúc ban đầu, nếu bạn không làm gì trong phút đầu tiên, chương trình khởi động sẽ được tự động bật sau đó. Đặc tính này có thể được loại bỏ bằng cách bấm một trong các phím trợ giúp màn hình ('help screen function keys').
- Nếu bạn bấm một trong các phím trợ giúp màn hình, thời gian trễ trong quá trình đọc màn hình trợ giúp từ đĩa sẽ không đáng kể.

Thông thường bạn chỉ cần bấm phím 'Enter' để khởi động. Hãy quan sát thông báo khởi động để kiểm tra liệu Linux có nhận diện được phần cứng của máy bạn hay không. Nếu không bạn phải khởi động lại ở chế độ "expert". Khi phần cứng của máy đã được nhận diện một cách xuôi xẻ, bạn có thể chuyển sang bước tiếp theo.

Bạn có thể vào chế độ "expert" bằng dòng khởi động sau đây:

```
boot: linux expert
```

Lưu ý: thông báo khởi động lúc ban đầu không chứa thông tin gì về SCSI hoặc card mạng bởi các thiết bị này sẽ được đọc trong quá trình khởi động.

Các lựa chọn sẽ được chuyển để xử lý.

Ví dụ, nếu muốn sử dụng toàn bộ RAM trong hệ thống 128MB, hãy vào:

```
boot: linux mem=128M
```

Sau khi đã chọn một số lựa chọn ('options'), hãy bấm 'Enter' để khởi động theo những lựa chọn đó.

Intel: bộ Red Hat Linux/Intel CD-ROM cũng có thể được dùng để khởi động trong các máy cho phép khởi động từ ổ CD-ROM. Nhưng nếu máy tính của bạn không cho phép khởi động từ ổ CD-ROM, bạn cũng có thể dùng cách khác để cài đặt mà không dùng đĩa khởi động. Cách sau đây chỉ dành cho các máy 'Intel-based'.

Nếu máy có cài đặt MS-DOS, bạn có thể khởi động trực tiếp từ ổ CD-ROM mà không cần tới đĩa khởi động bằng các dòng lệnh sau (coi ổ CD ở ổ D:):

```
C:\> d:
```

```
D:\> cd \dosutils
```

```
D:\dosutils> autoboot.bat
```

Cách này sẽ không áp dụng được khi làm việc dưới chế độ 'DOS window' do file 'autoboot.bat' chỉ được thực hiện dưới DOS như một hệ duy nhất.

Bạn sẽ phải sử dụng đĩa khởi động trong trường hợp bạn không thể khởi động trực tiếp từ ổ CD-ROM cũng như tự động ở 'DOS-based'.

P2.3. Lựa chọn phương pháp cài đặt

Tiếp theo, bạn có thể cài đặt Red Hat Linux theo một trong 5 phương pháp sau đây:

- **CD-ROM**

Nếu bạn có ổ CD-ROM và CD-ROM 'Red Hat Linux'. Để cài đặt cần có 1 đĩa khởi động, 1 CD-ROM khởi động được hoặc 1 đĩa khởi động PCMCIA.

- **Đĩa cứng**

Nếu bạn đã sao toàn bộ các file của bộ Red Hat Linux vào ổ cứng. Để cài đặt cần có 1 đĩa khởi động, hoặc 1 đĩa khởi động PCMCIA.

- **NFS**

Nếu bạn đang cài đặt từ 1 'NFS server' có 'Red Hat Linux' CD-ROM hoặc 1 phiên bản của Red Hat Linux. Để cài đặt cần 1 'network' hoặc 1 đĩa khởi động PCMCIA.

- **FTP**

Nếu bạn đang cài đặt trực tiếp từ 1 'FTP server'. Để cài đặt cần 1 'network' hoặc 1 đĩa khởi động PCMCIA.

- **HTTP**

Nếu bạn đang cài đặt trực tiếp từ 1 'HTTP Web server'. Để cài đặt cần 1 'network' hoặc 1 đĩa khởi động PCMCIA.

Phương pháp cài đặt qua CD-ROM sẽ được trình bày cụ thể trong phần tiếp theo.

P3. CÀI ĐẶT QUA GIAO DIỆN ĐỒ HỌA (CD-ROM)

Trong phần này ta sẽ xem xét việc cài đặt theo các bước sau:

- Bắt đầu cài đặt
- Lựa chọn ngôn ngữ
- Cấu hình bàn phím
- Cấu hình chuột
- Màn hình 'Welcome to Red Hat Linux'
- Cài đặt đường dẫn
- Phân hoạch tự động ('Automatic Partitioning')
- Phân hoạch hệ thống ('Partitioning your System')
- Chọn mẫu bảng FAT ('Choose Partitions to Format')

- Cài đặt LILO
- Cấu hình mạng (Network Configuration)
- Cấu hình thời gian (Time Zone Configuration)
- Đăng kí tài khoản (Account Configuration)
- Đăng kí sử dụng (Authentication Configuration)
- Lựa chọn các nhóm gói (Package Group Selection)
- Công cụ cấu hình GUI X (GUI X Configuration Tool)
- Chuẩn bị cài đặt
- Cài đặt các gói (Installing Packages)
- Tạo đĩa khởi động
- Kết thúc cài đặt (Installation complete)

1. Bắt đầu cài đặt

Để cài đặt Red Hat Linux từ CD-ROM, hãy chọn “CD-ROM” rồi chọn OK. Sau đó đưa đĩa CD “Red Hat Linux” vào ổ CD-ROM, chọn OK và nhấn “Enter”.

Chương trình cài đặt sẽ dò hệ thống và định danh ổ CD-ROM của máy bằng cách tìm một ‘integrated drive electronic’-IDE (được coi như một ATAPI) của ổ CD-ROM. Nếu tìm thấy, việc cài đặt sẽ được tiếp tục, nếu không, hệ thống sẽ hỏi loại (type) của thiết bị CD-ROM của máy và bạn có thể chọn một trong các loại sau:

- Loại SCSI

Được chọn nếu ổ CD-ROM được kèm theo một ‘adapter’ SCSI (Small Computer System Interface - SCSI). Chương trình cài đặt sẽ yêu cầu bạn mô tả để chọn thiết bị SCSI gần giống với ‘adapter’ của máy nhất hoặc ‘adapter’ SCSI sẽ được hệ thống phát hiện một cách tự động.

- Các loại khác

Được chọn nếu thiết bị CD-ROM không có cả IDE lẫn SCSI. Chương trình cài đặt sẽ đưa ra một danh sách các thiết bị để bạn lựa chọn.

☞ **Lưu ý:** một phần danh sách các tham số cho thiết bị CD-ROM được giới thiệu mục lục phần các tham số chung và mô-đun của tài liệu tham khảo ‘Official Red Hat Linux’. Nếu bạn có thiết bị IDE (ATAPI) CD-ROM mà chương trình cài đặt không tự tìm thấy được, bạn sẽ phải thực hiện lại chương trình cài đặt, rồi đưa vào linux hdX=cdrom, trong đó thay chữ X bằng một trong các chữ sau tùy theo IDE của máy:

- a - đối với bộ điều khiển IDE thứ nhất (First IDE controller), một ‘chủ’ (master)
- b - đối với bộ điều khiển IDE thứ nhất (First IDE controller), một ‘tớ’ (slave)

- c - đối với bộ điều khiển IDE thứ hai (Second IDE controller), một 'chủ' (master)
- d - đối với bộ điều khiển IDE thứ hai (Second IDE controller), một 'tớ' (slave)

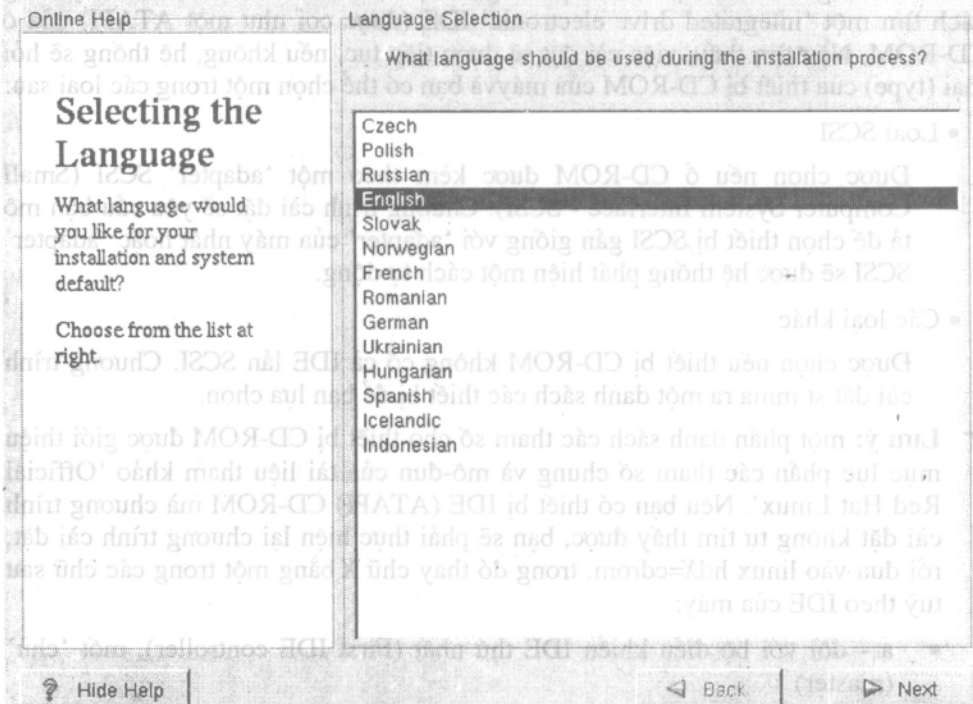
(Nếu máy của bạn có các bộ điều khiển IDE thứ 3, thứ 4, bạn sẽ nối chúng với các chữ cái tương ứng theo thứ tự chữ cái từ IDE này tới IDE khác và từ 'master' đến 'slave')

Sau khi ổ CD-Rom đã được định dạng xong, chương trình sẽ yêu cầu bạn chèn đĩa CD 'Red Hat Linux' vào ổ CD-ROM, rồi chọn OK, hộp thoại tiếp theo sẽ hiện ra một lát sau đó.

Khi đã khởi động xong, chương trình cài đặt sẽ sang phần lựa chọn ngôn ngữ. Nhưng nếu bạn muốn dừng việc cài đặt, bạn có thể rút đĩa khởi động ra và khởi động lại hệ thống.

Bạn cũng có thể huỷ bỏ việc cài đặt ở bất kì thời điểm nào trước khi bạn cấu hình X. Còn khi X đã được cấu hình xong, hệ thống sẽ ghi các 'partitions' và cài đặt các 'packages' (gói).

2. Lựa chọn ngôn ngữ



Hình P1. Lựa chọn ngôn ngữ

Bạn hãy sử dụng chuột để chọn ngôn ngữ ngầm định được sử dụng cho hệ thống và cho việc cài đặt (**hình P1**).

3. Cấu hình bàn phím

Trước tiên, hãy chọn mẫu giống với loại bàn phím của bạn nhất (**hình P2**). Nếu bạn không thể định dạng được kiểu bàn phím của máy, hãy chọn mẫu chung nhất cho hệ thống của bạn (ví dụ: Generic 101-key PC).

Tiếp theo, hãy chọn cách trình bày hợp lý nhất cho hệ thống (ví dụ: U.S. English). Ngoài ra, còn có một số lựa chọn khác cho bàn phím như việc lựa chọn các phím chết (Dead Keys option).

Bạn có thể kiểm tra cấu hình bàn phím trước khi click 'Next' để chuyển sang bước tiếp theo bằng cách sử dụng các chỗ để trống ở phía dưới của màn hình cấu hình. Nếu bạn muốn thay đổi kiểu của bàn phím sau khi đã cài đặt xong Red Hat Linux, hãy dùng lệnh '/usr/sbin/kbdconfig' hoặc gõ 'setup' ở dấu nhắc gốc (root prompt).

Online Help

Selecting the Keyboard

What kind of keyboard do you have?

Choose the model that best fits your system. If you can't find an exact keyboard match, choose the best *Generic* match (for example, Generic 101-key PC).

Then choose the layout type for your keyboard (for example, U.S. English).

If the language you've selected uses special

Keyboard Configuration

Model

- Generic 101-key (Intl) PC
- Generic 104-key PC**
- Generic 105-key (Intl) PC
- Japanese 106-key
- Keytronic FlexPro

Layout

- U.S. English
- U.S. English w/ deadkeys
- U.S. English w/ISO9995-3**
- United Kingdom

Variant

- Eliminate dead keys
- None**

Test your selection here:

? Hide Help

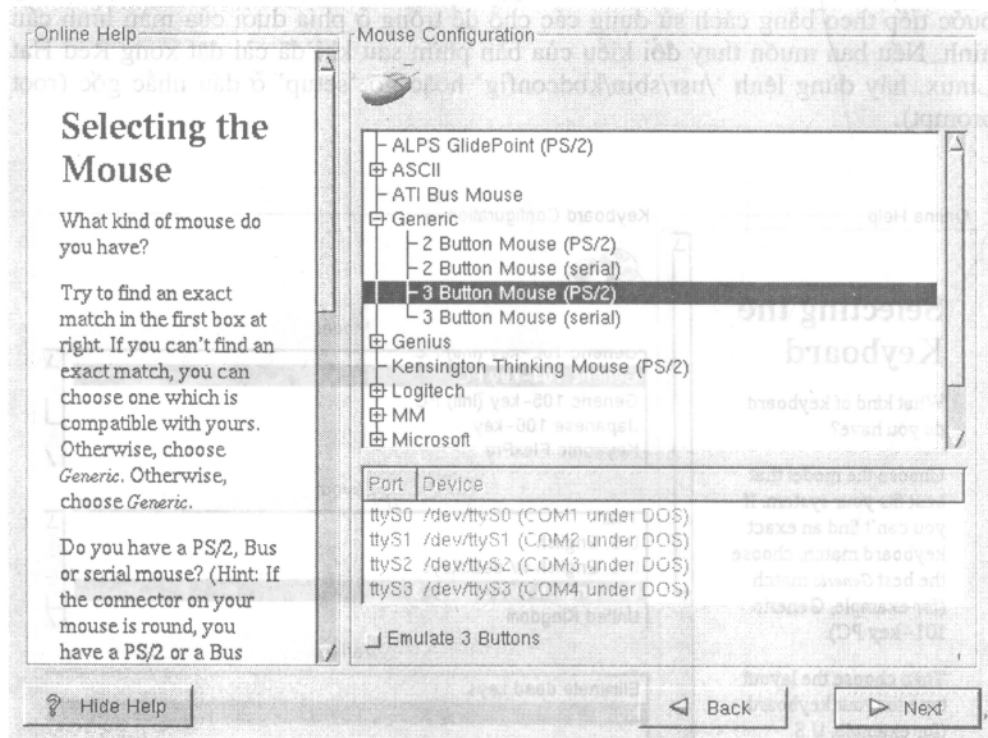
Back Next

Hình P2. Cấu hình bàn phím

4. Cấu hình chuột

Trước tiên, hãy chọn kiểu chuột (mouse type) cho hệ thống. Trong các mẫu được đưa ra, hãy chọn mẫu đúng hoặc tương thích nhất đối với hệ thống của bạn (hình P3).

Nếu chỗ phích cắm của chuột vào máy tính (phần giao diện của chuột) là hình chữ nhật nếu chuột là loại nối tiếp (serial mouse) và hình tròn nếu là loại PS/2. Nếu cài đặt Red Hat Linux trên máy xách tay, trong phần lớn trường hợp, mẫu chuột phải tương thích với PS/2. Nếu không thể tìm được kiểu chuột tương thích, hãy chọn các trong số các dạng chung (Generic entries) tùy theo số lượng button và giao diện của chuột.



Hình P3. Cấu hình chuột.

Nếu chuột ở dạng nối tiếp (serial mouse), bạn sẽ phải chọn đúng cổng và thiết bị cho nó.

Nếu muốn sử dụng chuột 2 buttons giống như chuột 3 buttons, hãy chọn check-box 'Emulate 3 Buttons'. Khi đó với chuột chỉ có 2 buttons, button ở giữa ("middle" button) được thể hiện bằng cách nhấn đồng thời chuột trái và chuột phải.

Nếu muốn thay đổi cấu hình chuột sau khi đã khởi động Red Hat Linux, hãy dùng lệnh '/usr/sbin/mouseconfig' từ dấu nhắc của shell (shell prompt).

Để chuột là 'left-handed', hãy bật thứ tự của các nút chuột. Điều này có thể được thực hiện sau khi đã khởi động Red Hat Linux bằng cách gõ 'gpm -B 321' tại dấu nhắc của shell.

5. Màn hình giới thiệu Red Hat Linux

Màn hình giới thiệu Red Hat Linux như sau (hình P4):

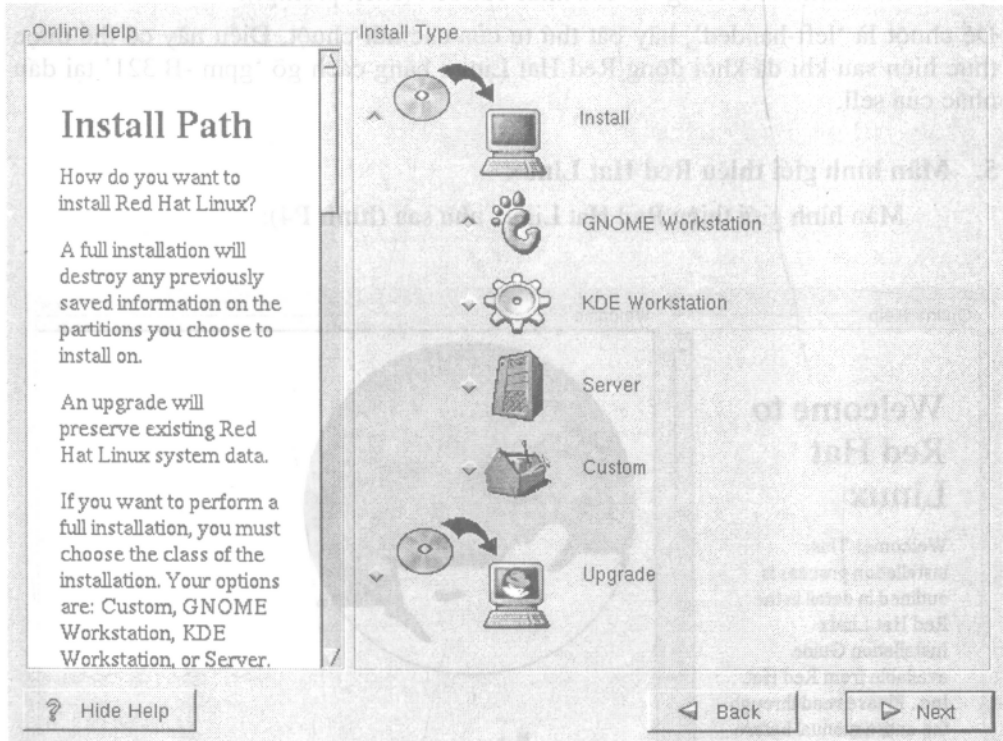


Hình P4. Màn hình giới thiệu Red Hat Linux

Lưu ý: nút trợ giúp ẩn (Hide Help button) nằm ở góc trái bên dưới của màn hình. Màn hình trợ giúp luôn được mở theo mặc định nhưng có thể thu nhỏ nó bằng cách nhấn vào nút 'Hide Help'.

6. Chọn kiểu cài đặt

Hãy chọn chức năng cài đặt hoặc nâng cấp (hình P5).



Hình P5. Chọn kiểu cài đặt.

P4. CÀI ĐẶT

Bạn có thể cài đặt Red Hat Linux trên một phiên bản khác của Linux hoặc không.

Warning (Cảnh báo)

Việc cài đặt Red Hat Linux trên một phiên bản khác của Linux (có cả Red Hat Linux) sẽ hủy bỏ mọi thông tin (files hoặc dữ liệu) được tạo ra trong lần cài đặt trước. Để tránh mất các thông tin quan trọng trong khi không tạo bản 'backup' riêng, thay vì chọn 'full installation', bạn nên chọn chức năng nâng cấp (xem phần *Upgrading*).

Nếu chọn 'full installation', bạn sẽ có các lựa chọn cài đặt sau đây: Custom (custom-class), GNOME Workstation hoặc KDE Workstation (workstation-class), và Server (server-class).

Nếu chọn 'workstation-class' hay 'server-class', việc cài đặt sẽ được thực hiện một cách tự động và một số bước trung gian bị bỏ qua. Nếu chọn 'custom-class', bạn sẽ kiểm soát toàn bộ các gói (packages) sẽ được cài đặt cũng như tự quyết định có sử dụng LILO để khởi động hệ thống hay không.

Nếu chọn 'workstation-class', bạn có thể chọn để cài đặt 'X Window' và 'desktop manager'.

Caution (khuyến cáo)

Nếu chọn 'workstation-class', mọi thông tin trong tất cả các phân hoạch có liên quan tới Linux (Linux-related partitions) ở mọi thiết bị cứng của máy sẽ đều bị xoá.

Nếu muốn hệ thống của bạn hoạt động như một 'Linux-based server', hãy chọn 'server-class', và bạn sẽ không phải thao tác nhiều trong quá trình cài đặt.

Caution

Nếu chọn 'server-class', mọi thông tin trong tất cả các phân hoạch có liên quan tới Linux hoặc không (both Linux and non-Linux) ở mọi thiết bị cứng của máy sẽ đều bị xoá.

Nâng cấp (Upgrading)

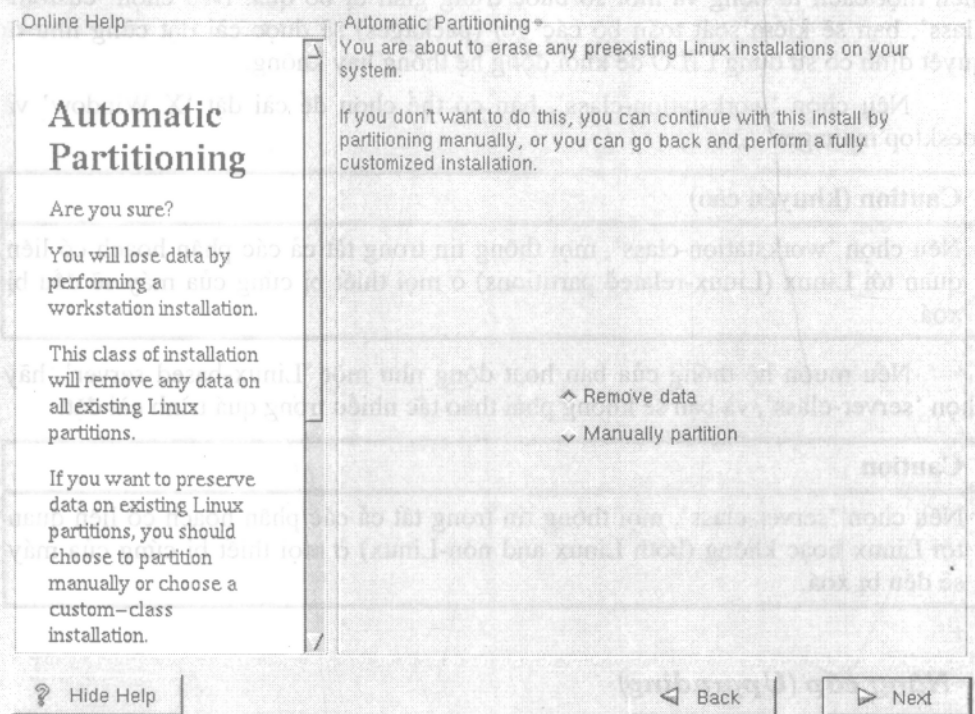
Từ các phiên bản Red Hat Linux (từ version 2.0 trở đi) dựa trên kỹ thuật RPM (Redhat Package Management), ta có thể nâng cấp chúng lên version 6.1. Quá trình nâng cấp sẽ lưu các files cấu hình cũ bằng cách thêm vào tên của nó phần mở rộng '.rpmsave' (ví dụ: sendmail.cf.rpmsave) và lưu các ghi chép lại (log of the actions) vào file '/tmp/upgrade.log'. Khi các phần mềm được nâng cấp thì khuôn dạng (format) của các files cấu hình cũng có thể bị thay đổi, do đó bạn nên so sánh các files cũ với các files mới trước khi tích hợp chúng với nhau.

☛ **Lưu ý:** một vài 'packages' đã được nâng cấp có thể yêu cầu một vài 'packages' khác cũng cần được cài đặt. Do đó, việc nâng cấp sẽ chú ý tới tính độc lập nhưng nó vẫn có thể cài đặt thêm một số 'packages' không có trên hệ thống cũ.

P4.1. Phân hoạch tự động

Màn hình chỉ xuất hiện nếu như trong chức năng cài đặt, ta chọn 'workstation-class' hoặc 'server-class'.

Tại đây, bạn có thể chọn 'Continue' để tiếp tục chương trình cài đặt, chọn để tự phân hoạch bằng tay, hoặc nhấn 'Back' để quay trở lại và chọn phương pháp cài đặt khác (hình P6).



Hình P6. Phân hoạch tự động

Nếu không muốn mất một phần hay toàn bộ dữ liệu, hãy chọn cách phân hoạch bằng tay, hoặc chọn phương pháp cài đặt khác.

☞ Lưu ý:

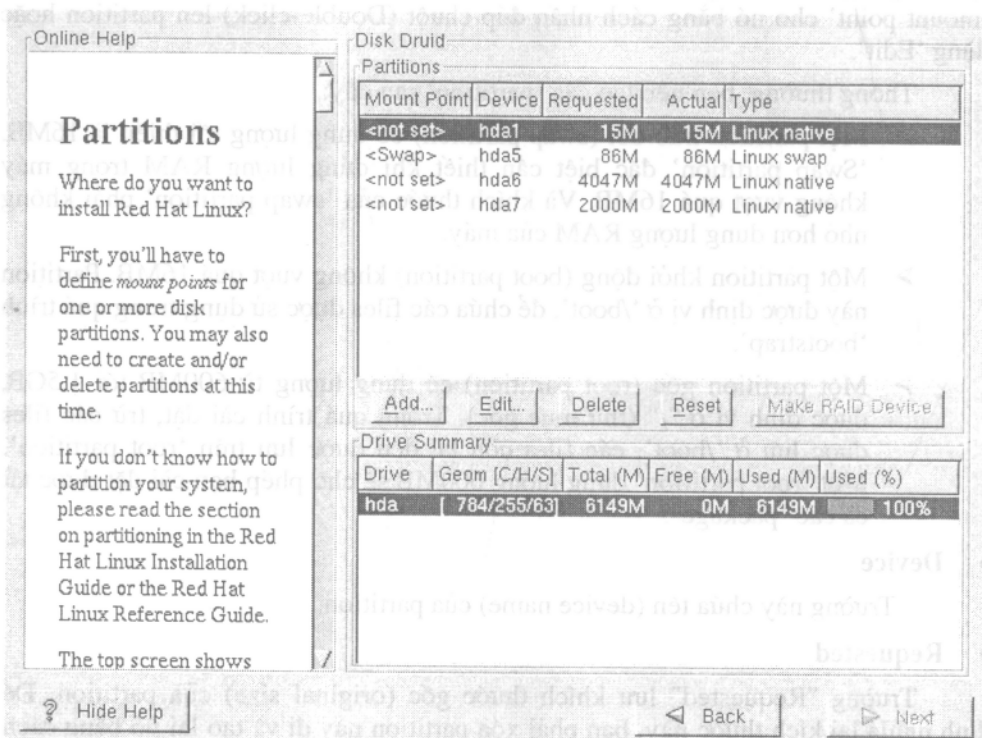
- không chọn 'workstation-class' hoặc 'server-class' nếu không muốn cài đặt Red Hat Linux trên 'master boot record' (MBR) hoặc nếu muốn sử dụng một hệ điều khiển khởi động (boot manager) khác với LILO.
- không chọn 'server-class' nếu muốn giữ lại một OS khác đã được cài đặt trên máy.

P4.2. Phân hoạch hệ thống

Trong trường hợp bạn đang cài đặt với lựa chọn 'workstation' hoặc 'server' nhưng không chọn phương pháp phân hoạch bằng tay, việc cài đặt sẽ bỏ qua các bước trong phần này để sang phần *cấu hình thời gian (Time Zone Configuration)*.

Ở bước phân hoạch hệ thống, bạn sẽ phải thông báo cho chương trình cài đặt biết Red Hat Linux sẽ được cài đặt trên các 'disk partition' nào. Và lúc này, bạn cũng có thể tạo thêm hoặc xóa bớt đi một số 'partition' (**hình P7**).

🔧 **Lưu ý:** bạn có thể xem phần phân hoạch để biết thêm chi tiết về phân hoạch đĩa; ở mức tối thiểu, bạn phải có một 'root partition' và một 'swap partition' có kích thước tối thiểu 16 MB.



Hình P7. Phân hoạch hệ thống.

Công cụ phân hoạch của Red Hat Linux 6.1 là *Disk Druid*. Trong phiên bản này, 'Disk Druid' sử dụng giao diện đồ họa.

Alpha, SPARC: khi sử dụng 'Disk Druid' trên máy Alpha hoặc SPARC, hãy xem phần 'Official Red Hat Linux Alpha/SPARC Installation Addendum' để biết thêm chi tiết.

a) Các trường của partition

Mỗi dòng của mục "Partitions" sẽ đại diện cho một 'disk partition', và nó gồm 5 trường sau:

- **Mount Point**

Một 'mount point' là một vị trí trong cây thư mục với dung lượng thực tế của nó. Trường này chỉ ra vị trí mà 'partition' sẽ được định vị. Nếu một 'partition' tồn tại trong thực tế nhưng chưa được khai báo ("not set"), bạn sẽ phải định nghĩa 'mount point' cho nó bằng cách nhấn đúp chuột (Double-click) lên partition hoặc dùng 'Edit'.

Thông thường, bạn nên tạo các 'partition' sau đây:

- Một partition trao đổi (swap partition) có dung lượng tối thiểu là 16MB. 'Swap partition' đặc biệt cần thiết khi dung lượng RAM trong máy không vượt quá 16MB. Và kích thước của 'swap partition' phải không nhỏ hơn dung lượng RAM của máy.
- Một partition khởi động (boot partition) không vượt quá 16MB. Partition này được định vị ở '/boot', để chứa các files được sử dụng trong quá trình 'bootstrap'.
- Một partition gốc (root partition) có dung lượng từ 600MB tới 1.5GB, được định vị ở '/' (thư mục gốc). Trong quá trình cài đặt, trừ các files được lưu ở '/boot', các files còn lại đều được lưu trên 'root partition'. Một 'root partition' dung lượng 600MB sẽ cho phép bạn cài đặt được tất cả các 'package'.

- **Device**

Trường này chứa tên (device name) của partition.

- **Requested**

Trường "Requested" lưu kích thước gốc (original size) của partition. Để định nghĩa lại kích thước này, bạn phải xóa partition này đi và tạo lại nó bằng cách sử dụng nút 'Add button'.

- **Actual**

Trường "Actual" chỉ ra dung lượng nhớ hiện tại được cấp cho partition.

- **Type**

Trường này chỉ kiểu của partition (ví dụ: Linux Native hoặc DOS).

b) Các vấn đề khi thêm một partition

Nếu muốn thêm một partition mới mà 'Disk Druid' không thể thực hiện được yêu cầu này, hãy tìm các partition chưa được cấp phát (cùng lí do chưa được cấp phát của nó) trong danh sách trong 'dialog box', mục "Partitions", phần "Unallocated Requested Partition". Một partition không được cấp phát thường chủ yếu do việc thiếu bộ nhớ.

Đối với một partition không được cấp phát, bạn phải di chuyển nó sang một thiết bị khác (còn đủ bộ nhớ để cấp cho nó) để thay đổi lại kích thước của partition này cho phù hợp với thiết bị trước đó, hoặc xóa toàn bộ partition này đi. Điều này được thực hiện bằng cách sử dụng nút 'Edit' hoặc kích đúp chuột lên partition.

c) Tóm tắt về các thiết bị (Drive Summaries)

Mỗi một dòng của "Drive Summaries" sẽ giới thiệu các thông tin của một đĩa cứng trong hệ thống, nó bao gồm các trường sau:

- **Drive:** chứa tên của đĩa cứng.
- **Geom [C/H/S]:** trường này chứa 3 số giới thiệu về số cylinders, số đầu từ (heads) và số sectors của đĩa.
- **Total:** Trường "Total" nêu tổng dung lượng bộ nhớ của đĩa.
- **Free:** Trường "Free" nêu lên dung lượng bộ nhớ còn trống của đĩa.
- **Used:** chỉ dung lượng bộ nhớ đã được sử dụng trên đĩa theo 'megabytes' (MB) và theo phần trăm.

d) Các nút điều khiển trong Disk Druid

Các nút điều khiển trong 'Disk Druid' được sử dụng để thêm, bớt, thay đổi các thuộc tính của partition, cho phép chấp nhận các thay đổi do bạn tạo ra hoặc cho phép rời khỏi 'Disk Druid'. Chức năng của từng nút như sau:

- **Add:** được dùng để tạo một partition mới. Hãy điền thông tin vào các mục trong hộp thoại được mở ra.
- **Edit:** cho phép thay đổi các thuộc tính của partition 'highlighted' (được nhấn mạnh) trong mục "Partitions". Các mục trong hộp thoại 'Edit Partition' có thể thay đổi phụ thuộc vào thông tin của partition đã được ghi lên đĩa.
- **Delete:** cho phép xóa partition được chọn trong mục 'Current Disk Partitions'.
- **Reset:** sử dụng để phục hồi 'Disk Druid' trở về trạng thái ban đầu của nó, và huỷ bỏ mọi thay đổi do bạn đã tạo ra.
- **Make RAID Device:** cho phép tạo sự dư thừa cho 0 hoặc tất cả các 'disk partitions' (chỉ nên dùng nếu bạn đã hiểu rất rõ về RAID).

e) Thêm partition

Hộp thoại sau sẽ xuất hiện khi chọn nút 'Add' để thêm partition mới (hình P8):

Mount Point:

Size (Megs):

☐ Grow to fill disk?

Partition Type: Linux Native

Allowable Drives:

hda

Hình P8. Thêm partition

- Mount Point: đưa vào thư mục nơi partition định vị (ví dụ: '/' đối với 'root partition'; '/usr' đối với '/usr partition'; ...); hoặc chọn trong danh sách có sẵn 'mount point' tương ứng với partition.
- Size (Megs): nhập kích thước (số MB) của partition (giá trị ngầm định: 1 (MB)).
- Grow To Fill: kích thước được đưa vào của partition sẽ là kích thước nhỏ nhất khi 'check box' này được tích, và là kích thước đúng trong trường hợp ngược lại.
- Allowable Type: hãy dùng chuột để chọn kiểu (type) của partition trong số các kiểu được liệt kê (ví dụ: Linux Native, DOS).
- Allowable Drives: chỉ khi 'check box' này được tích thì partition cần tạo mới được tạo ra trên đĩa cứng được chọn (hard disk's box highlighted). Do đó, bạn có thể hướng dẫn 'Disk Druid' định vị các partition hoặc để 'Disk Druid' tự thực hiện.
- Ok: chọn và ấn 'Enter' sẽ cho phép tạo partition với các thuộc tính đã được thiết lập.
- Cancel: chọn và ấn 'Enter' nếu không muốn tạo partition.

f) Sửa đổi các partition

Để sửa đổi một partition, hãy chọn "Edit" hoặc nhấn đúp chuột lên partition hiện có.

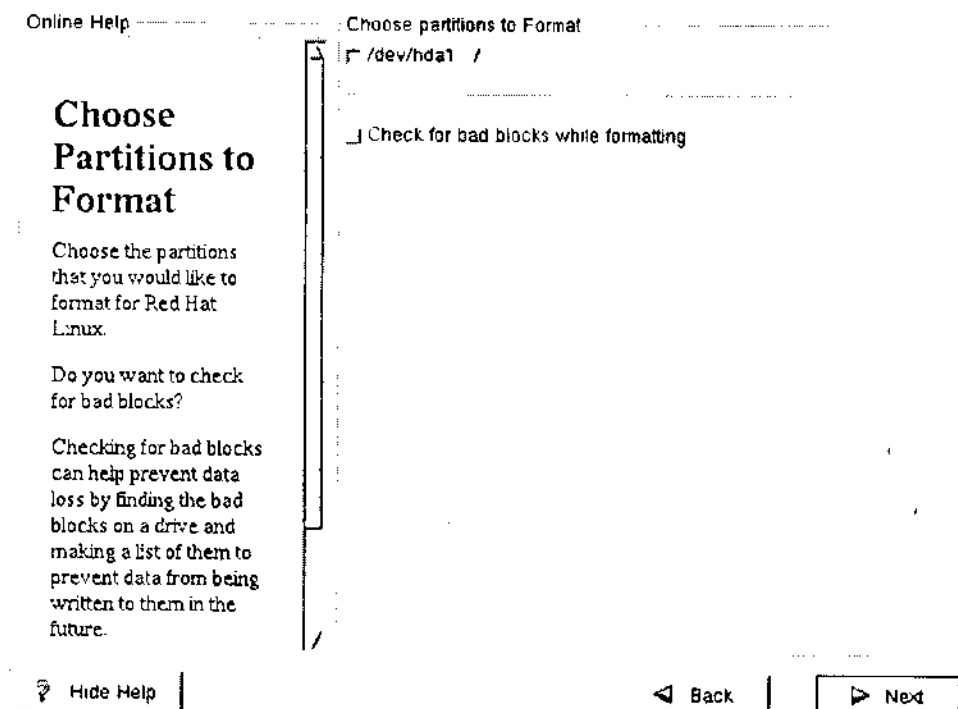
- **Lưu ý:** với một partition đã tồn tại trên đĩa cứng, ta chỉ có thể thay đổi 'mount point' của nó; nếu muốn tạo các thay đổi khác, ta phải xóa partition này và tạo lại partition khác.

g) Xóa một partition

Hãy chọn partition cần xóa trong số các partition ở mục "Partitions", rồi nhấn đúp (double-click) vào nút "Delete".

P4.3. Chọn các partition để format

Trừ các partition có chứa những thông tin quan trọng như '/home' hoặc '/usr/local', còn mọi partition vừa được tạo ra hoặc lưu những thông tin mà ta không cần nữa đều nên được format lại (**hình P9**).



Hình P9. Chọn các partition để format.

Nếu muốn kiểm tra các block bị hỏng trong quá trình format từng file hệ thống, hãy chọn 'check for bad blocks'. Lựa chọn này có thể giúp ngăn cản việc mất dữ liệu bị lưu tại các block bị hỏng, đồng thời tạo một danh sách các block bị hỏng để tránh sử dụng lại chúng sau này.

P4.4. Cài đặt LILO

Bước này chỉ được thực hiện trong trường hợp không cài đặt với lựa chọn 'workstation-class' hay 'server-class'.

Để khởi động hệ Red Hat Linux, bạn nên cài đặt LILO (LIⁿux LO^ader), và có thể thực hiện nó trên một trong 2 vị trí sau:

- *Bản ghi khởi động chủ (Master boot record - MBR)*: Bạn nên cài đặt LILO trên MBR, trừ khi MBR vừa khởi động một hệ tải (operating system loader) khác (như 'System Commander' hoặc 'Boot Manager' của hệ điều hành đa nhiệm - OS/2). MBR là vùng đặc biệt được tự động tải bởi BIOS, và cũng là vùng mà tại đó LILO có thể nắm giữ quyền kiểm soát tiến trình khởi động. Nếu LILO được cài đặt trên MBR, thì khi khởi động, LILO sẽ hiện diện ở 'boot: prompt'. Sau đó, bạn có thể khởi động Red Hat Linux hay bất kì hệ điều hành nào khác mà bạn đã cấu hình LILO để khởi động.
- *Sector đầu tiên của partition gốc*: được áp dụng trong trường hợp bạn đang dùng một 'boot loader' khác (ví dụ: 'Boot Manager' của OS/2). Khi đó, loader này sẽ nắm quyền kiểm soát trước tiên, và bạn có thể cấu hình để nó khởi động LILO (rồi LILO sẽ khởi động Red Hat Linux).

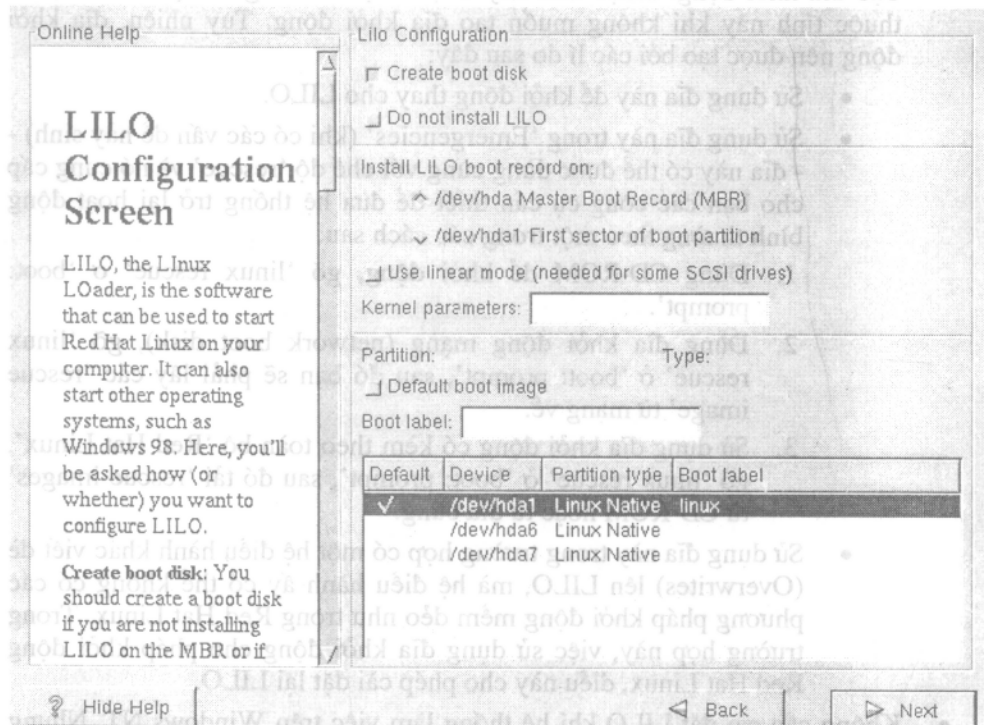
Hãy chọn nơi trên hệ thống mà bạn sẽ cài đặt LILO (**hình P10**). Nên chọn MBR nếu hệ thống chỉ dùng duy nhất Red Hat Linux. Nếu hệ thống đã cài Win95/98, thì có thể cài đặt LILO trên MBR sau cho LILO có thể khởi động cả hai hệ điều hành.

Với máy cài Windows NT, thì LILO nên được cài đặt ở '*Sector đầu tiên của partition gốc*'. 'Operating system loader' của Windows NT sẽ nằm ở MBR, và bạn chỉ có thể chọn LILO thông qua *đĩa khởi động* (boot disk). Có thể biết thêm thông tin về cài đặt LILO và NT qua trang Web sau: <http://www.linuxdoc.org/HOWTO/mini/Linux+NT-Loader.html>.

Warning

Nếu không cài đặt LILO, việc khởi động hệ Red Hat Linux sẽ không thể được thực hiện bằng cách trực tiếp (ví dụ phải sử dụng một đĩa khởi động - *boot diskette*).

Nếu thiết bị là một SCSI và bạn gặp phải lỗi truy nhập thiết bị, nên check chế độ 'Use linear' nếu máy truy nhập ổ cứng dưới chế độ LBA.



Hình P10. Đặt cấu hình LILO

Có thể thêm các lựa chọn ngầm định vào yêu cầu khởi động LILO bằng cách đưa chúng vào các trường tham số của 'kernel' (hạt nhân). Các lựa chọn được đưa vào sẽ được chuyển tới 'Linux kernel' ở tất cả các thời điểm khởi động LILO.

Partition khởi động được (Bootable Partition) — mọi 'bootable partition' kể cả các partition được sử dụng bởi các hệ điều hành khác, đều được liệt kê. Trừ partition có lưu các files hệ thống gốc (ở root) của hệ Red Hat Linux, các partition khác có thể được thêm hoặc thay đổi *nhãn khởi động* (ở cột "Boot label") bằng cách click chuột lên partition để chọn rồi thay đổi nhãn của partition.

Lưu ý: Nội dung của cột "Boot label" được bạn đưa vào từ 'boot: prompt' của LILO theo thứ tự khởi động của hệ điều hành được chọn. Bạn có thể ấn phím 'Tab' từ 'boot: prompt' của LILO để xem danh sách các nhãn khởi động đã được định nghĩa.

* Cấu hình LILO :

Ta có các lưu ý sau:

- Tạo đĩa khởi động: thuộc tính này được chọn theo ngầm định. Có thể bỏ thuộc tính này khi không muốn tạo đĩa khởi động. Tuy nhiên, đĩa khởi động nên được tạo bởi các lí do sau đây:
 - Sử dụng đĩa này để khởi động thay cho LILO.
 - Sử dụng đĩa này trong 'Emergencies' (khi có các vấn đề nảy sinh) - đĩa này có thể được dùng cùng với chế độ 'rescue' và nó cung cấp cho bạn các công cụ cần thiết để đưa hệ thống trở lại hoạt động bình thường theo một trong các cách sau:
 1. Dùng CD-ROM để khởi động, gõ 'linux rescue' ở 'boot: prompt'.
 2. Dùng đĩa khởi động mạng (network boot disk), gõ 'linux rescue' ở 'boot: prompt', sau đó bạn sẽ phải lấy các 'rescue image' từ mạng về.
 3. Sử dụng đĩa khởi động có kèm theo toàn bộ 'Red Hat Linux', gõ 'linux rescue' ở 'boot: prompt', sau đó tải 'rescue images' từ CD-ROM hoặc từ đĩa cứng.
 - Sử dụng đĩa này trong trường hợp có một hệ điều hành khác viết đè (Overwrites) lên LILO, mà hệ điều hành ấy có thể không có các phương pháp khởi động mềm dẻo như trong Red Hat Linux. Trong trường hợp này, việc sử dụng đĩa khởi động cho phép khởi động Red Hat Linux, điều này cho phép cài đặt lại LILO.
- Không cần cài đặt LILO khi hệ thống làm việc trên Windows NT. Nhưng cần tạo đĩa khởi động bởi nó mới cho bạn khởi động được Linux.

* Các lựa chọn cho LILO (Alternatives to LILO) :

Có một số lựa chọn sau đây khi bạn không muốn dùng LILO để khởi động hệ điều hành Red Hat Linux:

• Boot Disk:

Sử dụng đĩa khởi động được tạo ra bởi chương trình cài đặt.

• LOADLIN:

Linux có thể được tải từ MS-DOS với điều kiện cần có một phiên bản hợp lệ của 'Linux kernel' (và bộ nhớ RAM nếu hệ thống có 'SCSI adapter') trên một partition MS-DOS. để thực hiện điều này, bạn cần khởi động Red Hat Linux từ LILO trên một đĩa nhỏ, sau đó copy 'kernel' vào partition MS-DOS. Có thể lấy LOADLIN từ địa chỉ:

<ftp://metalab.unc.edu/pub/Linux/system/boot/dualboot/> và một số site được kết nối tới nó.

- **SYSLINUX:**

Các chương trình của MS-DOS tương tự như của LOADLIN, và bạn có thể tìm ở địa chỉ: <ftp://metalab.unc.edu/pub/Linux/system/boot/loaders/> và một số site được kết nối tới nó.

- **Một số "bootloaders" được thương mại hóa:**

Ví dụ: 'System Commander', 'Partition Magic' đều cho phép khởi động Linux, nhưng chúng vẫn cần LILO phải được cài đặt ở 'root partition'.

*** Các bản mạch chính SMP (SMP Motherboards) và LILO :**

Khi chương trình cài đặt nhận dạng được 'SMP motherboards', nó sẽ tự động tạo ra 2 phần tử 'lilo.conf', mà một được gọi là 'linux' và phần tử kia là 'linux-up'. Phần tử 'linux' sẽ được khởi động một cách ngầm định, nhưng bạn có thể thay thế nó bởi 'linux-up' khi gặp vấn đề với 'SMP kernel', và khi đó, bạn vẫn giữ được tất cả các chức năng như trước đây nhưng chỉ được làm việc với một bộ xử lí đơn.

P4.5. Cấu hình mạng

Online Help

Network Configuration

If you have a network card, you can set up your networking information. Otherwise, click Next to proceed.

Choose your device type and whether you would like to configure using DHCP. If you have multiple ethernet devices, each device screen will keep the information you have given to it. You can switch between device screens, for example eth0 and eth1: the

Network Configuration

eth0

☐ Configure using DHCP

☒ Activate on boot

IP Address:

Netmask:

Network:

Broadcast:

Hostname:

Gateway:

Primary DNS:

Secondary DNS:

Tertiary DNS:

? Hide Help

< Back

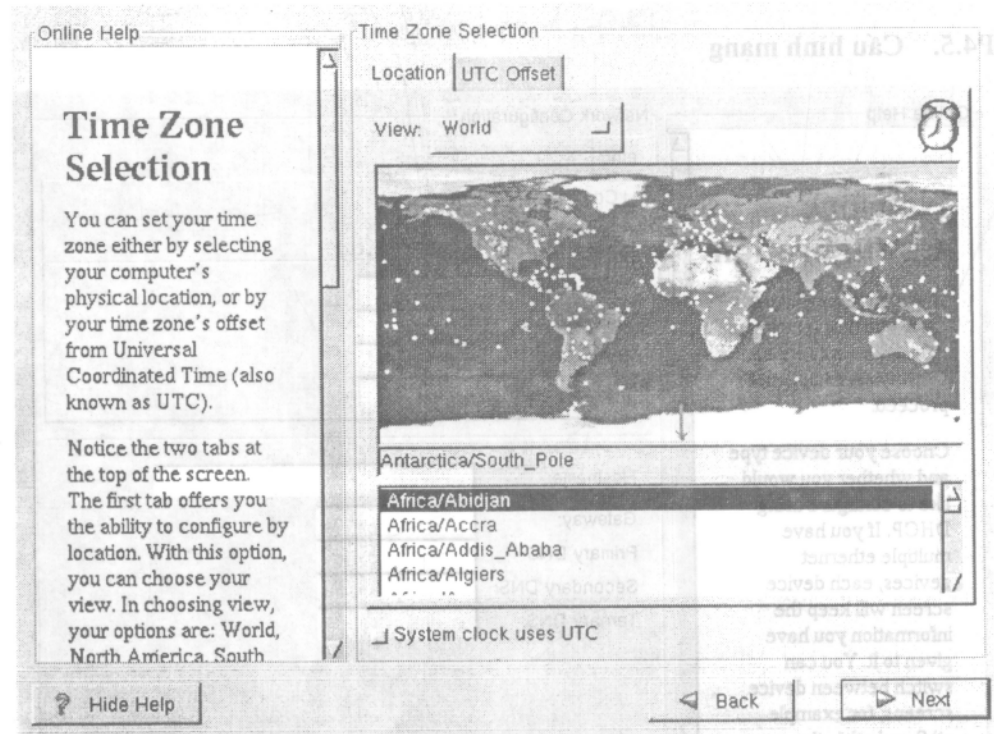
Next >

Hình P11. Cấu hình mạng

Trước tiên, hãy chọn kiểu thiết bị, và có sử dụng hay không DHCP. Nếu có nhiều thiết bị ethernet, bạn có thể chuyển đổi (switch) giữa chúng, và mỗi thiết bị này sẽ được đặc trưng bởi các thông tin mà bạn cung cấp cho chúng. Nếu chọn 'Activate on boot', giao diện mạng sẽ được bật khi khởi động. Có thể liên hệ với quản trị mạng 'network administrator' để biết thêm chi tiết về DHCP. Tiếp theo, đưa vào địa chỉ IP, Netmask, Network, và số hiệu 'Broadcast' (có thể liên hệ với quản trị mạng 'network administrator' để biết thêm chi tiết). Sau đó đưa tên của hệ thống để hệ thống được coi như một 'localhost'. Cuối cùng, hãy đưa vào số hiệu của: Gateway, DNS cơ sở, DNS thứ 2 và thứ 3.

P4.6. Cấu hình thời gian (Time Zone Configuration)

Việc này được thực hiện bằng cách chọn vị trí (về mặt địa lý) của máy hoặc độ bù của thời gian (time zone's offset) so với 'Universal Coordinated Time' (còn gọi là UTC).



Hình P12. Cấu hình thời gian

Phím tab đầu tiên cho phép cấu hình theo vị trí địa lý, còn phím tab thứ 2 cho phép dùng độ bù UTC (hình P12).

☛ **Chú ý:** hãy chọn UTC nếu 'System Clock' sử dụng UTC.

Sau khi khởi động xong Red Hat Linux, có thể thay đổi 'time zone configuration' của hệ thống bằng lệnh '/usr/sbin/timeconfig'.

P4.7. Đăng kí tài khoản (Account Configuration)

Trên màn hình 'Account Configuration', bạn có thể đặt 'root password', và các 'account' của những người sử dụng khi login (hình P13).

Account Configuration

Root Password:

Confirm:

Account Name:

Password: Password (confirm):

Full Name:

Account Name	Full Name

Online Help

Account Configuration

Enter a root password. The password must be at least six characters in length. Confirm the password. The "Next" button will become enabled once both entry fields match.

Now create a user account.

Enter a user account name. Then, create a password for that user account and confirm it. Finally, enter the full name of the account

Hình P13. Đăng kí tài khoản.

• Đặt 'Root Password'▶

'Root password' dành cho 'root user' và được sử dụng để log lần đầu tiên vào Red Hat Linux. Mật khẩu này phải gồm ít nhất 6 ký tự, làm cho bạn dễ nhớ mà

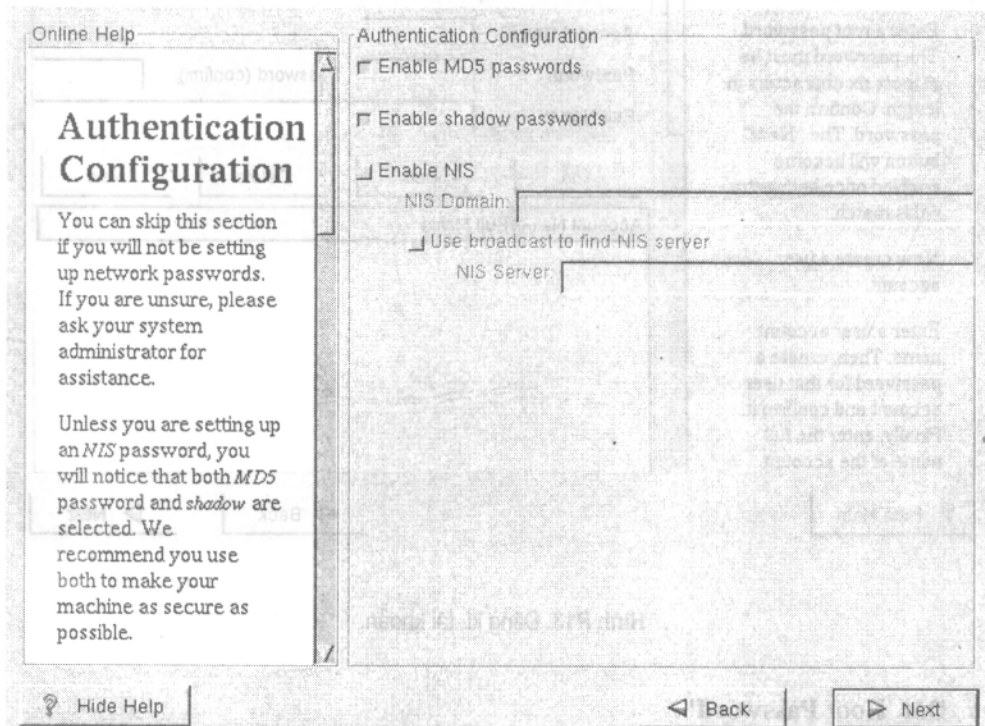
người ngoài khó tìm ra nó. Khi được gõ vào, nó sẽ không được hiện lên màn hình, bạn sẽ phải gõ lại nó lần thứ hai, và khi 2 mật khẩu này không trùng nhau, thì bạn sẽ phải đặt lại 'root password'.

☞ **Chú ý:** 'Root user' (hay 'superuser') có quyền truy nhập vào mọi nơi của hệ thống, nên thường chỉ login với tư cách 'root user' khi muốn bảo trì hoặc quản trị hệ thống.

• Đặt "User Accounts"

Có thể chọn để tạo các 'user account' trong quá trình cài đặt, còn khi chương trình đã được cài đặt xong thì chỉ 'root user' mới có quyền tạo chúng. Đưa vào tên của 'account', rồi đưa vào và khẳng định 'password' (mật khẩu) cho 'user account'. Tiếp theo đưa vào tên đầy đủ của 'user account' này rồi ấn 'Enter'. Bạn có thể chọn 'New' để đưa thêm 'user' mới, nhập các thông tin về 'user' này, rồi chọn nút 'Add' để đưa 'user' này vào danh sách các 'account'. Bạn cũng có thể chọn 'Edit' hoặc 'Delete' để thay đổi hoặc xóa các 'user account' khỏi danh sách các 'account'.

P4.8. Đăng kí quyền sử dụng (Authentication Configuration)



Hình P14. Đăng kí quyền sử dụng.

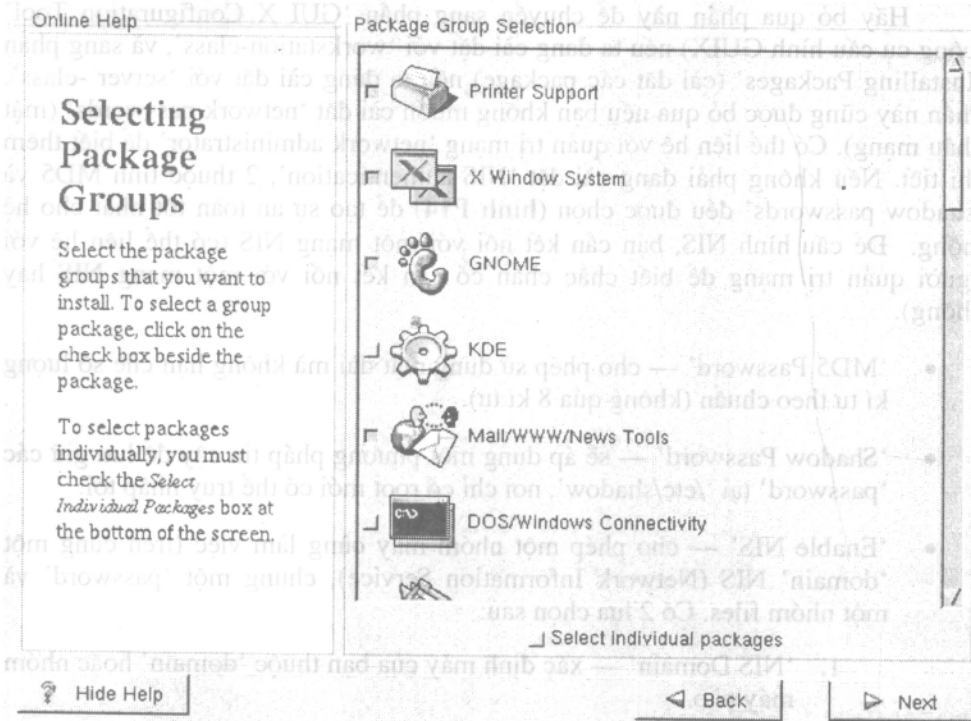
Hãy bỏ qua phần này để chuyển sang phần 'GUI X Configuration Tool' (công cụ cấu hình GUIX) nếu ta đang cài đặt với 'workstation-class', và sang phần 'Installing Packages' (cài đặt các package) nếu ta đang cài đặt với 'server-class'. Phần này cũng được bỏ qua nếu bạn không muốn cài đặt 'network passwords' (mật khẩu mạng). Có thể liên hệ với quản trị mạng 'network administrator' để biết thêm chi tiết. Nếu không phải đang cài đặt 'NIS authentication', 2 thuộc tính MD5 và 'shadow passwords' đều được chọn (**hình P14**) để tạo sự an toàn tốt nhất cho hệ thống. Để cấu hình NIS, bạn cần kết nối với một mạng NIS (có thể liên hệ với người quản trị mạng để biết chắc chắn có cần kết nối với một mạng NIS hay không).

- 'MD5 Password' — cho phép sử dụng mật dài mà không hạn chế số lượng ký tự theo chuẩn (không quá 8 ký tự).
- 'Shadow Password' — sẽ áp dụng một phương pháp tin cậy để lưu giữ các 'password' tại '/etc/shadow', nơi chỉ có root mới có thể truy nhập tới.
- 'Enable NIS' — cho phép một nhóm máy cùng làm việc (trên cùng một 'domain' NIS (Network Information Service), chung một 'password' và một nhóm files. Có 2 lựa chọn sau:
 1. 'NIS Domain' — xác định máy của bạn thuộc 'domain' hoặc nhóm máy nào.
 2. 'NIS Server' — xác định 'NIS server' sẽ làm việc với hệ thống của bạn.

P4.9. Lựa chọn nhóm phần mềm (package)

Chương trình cho phép 'klik' vào các 'check box' để lựa chọn các thành phần (tức nhóm các package nào của cùng một chức năng, ví dụ: 'C Development', 'Networked Workstation', hoặc 'Web Server'), 'individual packages' (các package riêng lẻ), hoặc sự tổ hợp giữa chúng (**hình P15**).

Hãy chọn từng thành phần để cài đặt, hoặc có thể chọn 'Selecting Everything' sẽ cài đặt tất cả các 'package' của Red Hat Linux. Việc cài đặt tất cả các package sẽ cần tới 1.5GB trống trên đĩa. Việc chọn hộp 'Select Individual Packages' nằm ở phía dưới màn hình sẽ cho phép chọn các package riêng lẻ.



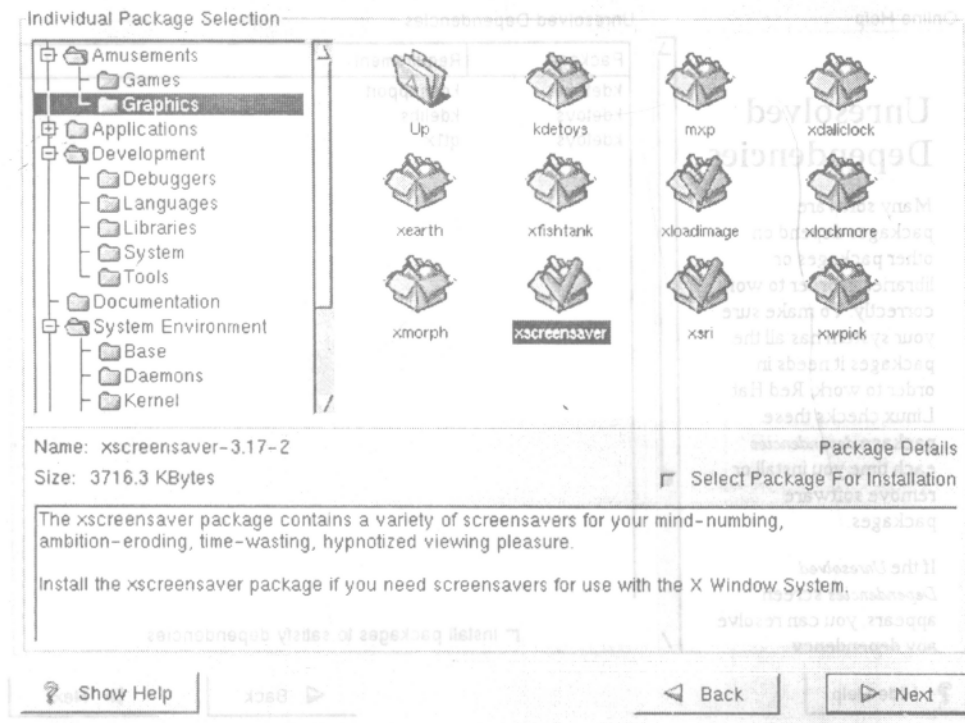
Hình P15. Lựa chọn nhóm package

a) Chọn các package riêng lẻ

Sau khi chọn xong các thành phần (components) cần cài đặt, bạn có thể dùng chuột để chọn hoặc xóa các package đơn lẻ được hiển thị trong danh sách các package của nhóm (hình P16).

Danh sách thư mục của các package được hiển thị phía bên trái màn hình. Khi nhấn đúp chuột để chọn một thư mục, thì danh sách các package trong thư mục này sẽ xuất hiện phía bên phải màn hình. Bạn có thể chọn từng package riêng lẻ theo 2 cách sau: nhấn đúp chuột trái, hoặc nhấn để 'highlight' package rồi nhấn nút 'Select Package For Installation' ở phía dưới. Các package được chọn để cài đặt sẽ được đánh dấu bởi màu đỏ. Các thông tin của package được 'highlight' cùng với tên và kích thước của nó sẽ xuất hiện ở phía dưới màn hình.

Chú ý: một số packages cơ sở (như 'kernel' và một vài thư mục) cần thiết trong mọi phiên bản Red Hat Linux luôn được chọn một cách ngầm định.

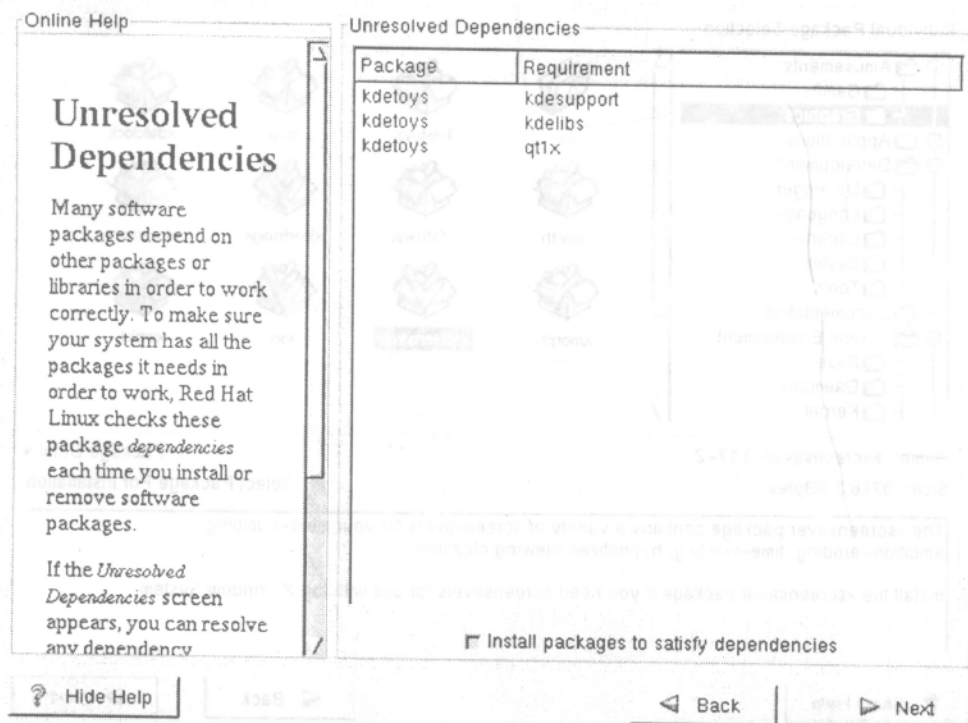


Hình P16. Chọn các package riêng lẻ.

b) Các phụ thuộc còn thiếu (Unresolved Dependencies)

Trong Red Hat Linux có một số package phụ thuộc vào nhau và cần cùng được cài đặt cho hệ thống. Do đó Red Hat Linux luôn kiểm tra sự phụ thuộc giữa các package mỗi khi bạn cài đặt hay gỡ bỏ ('remove') các package của phần mềm ('software packages'). Khi một package nào đó cần package khác mà bạn chưa chọn để cài đặt, thì chương trình cài đặt sẽ đưa ra danh sách các 'unresolved dependencies' và cho bạn một số lựa chọn để giải quyết chúng (hình P17).

Màn hình 'Unresolved Dependencies' chỉ xuất hiện khi thiếu một số package cần cho các package đã được chọn. Phía dưới danh sách các package bị thiếu có một 'check box' ('Install packages to satisfy dependencies') được ngầm định chọn. Khi rời khỏi màn hình này, chương trình cài đặt sẽ tự động thêm các package còn thiếu vào danh sách các package đã được chọn.



Hình P17. Các phụ thuộc còn thiếu.

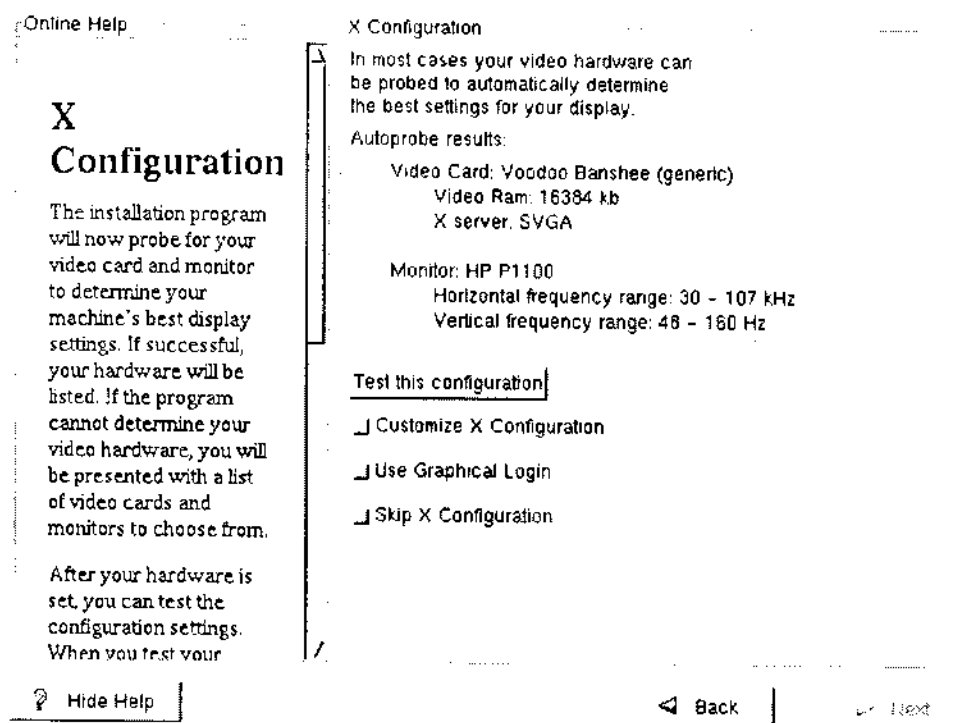
P4.10. Công cụ cấu hình GUI X

Nếu bạn không muốn cài đặt 'X Window', hãy bỏ qua phần này để chuyển sang phần cài đặt các package (Installing Packages). Trước tiên, chương trình 'Xconfigurator' sẽ cố gắng xác định kiểu card màn hình (video hardware) của máy (hình P18), nếu không được, Xconfigurator sẽ cho phép chọn từ danh sách các card video và 'monitors'.

Card video của máy sẽ không có trong danh sách có thể do XFree86 không chấp nhận. Khi đó, có thể chọn 'Unlisted Card' và cấu hình nó bằng cách đối sánh 'chipset' của card video của máy và của 'X servers' với nhau.

Tiếp theo, đưa thông tin về lượng bộ nhớ video của card video (có thể tra phần tài liệu kèm theo của card video của máy để có số liệu chính xác). Thông tin này phải chính xác để tránh việc hoạt động sai của 'XFree86 server'. 'Xconfigurator' sẽ đưa ra danh sách các 'clockchip' (chip đồng hồ) nếu như card video có 'clockchip'. Nên chọn 'No Clockchip Setting' để XFree86 tự động nhận dạng 'clockchip'. Hãy chọn 'Custom' nếu 'monitor' của máy không có trong danh sách, rồi chọn các thông tin về 'horizontal sync range' và 'vertical sync range'.

(phạm vi đồng bộ theo chiều ngang và dọc) của 'monitor' (xem tài liệu kèm theo 'monitor' của máy hoặc từ nhà cung cấp).



Hình P18. Công cụ cấu hình GUI X

Caution (cảnh báo)

'Monitor' của máy có thể bị tổn hại nếu 'monitor' được chọn vượt quá khả năng thực tế của nó.

Bạn có thể 'test' cấu hình của hệ thống (về độ phân giải và màu sắc) sau khi đã nhận dạng hoàn chỉnh phần cứng. Bạn có thể tự chọn cấu hình X bằng cách nhấn nút 'Customize X Configuration', rồi chọn màu và độ phân giải mong muốn. Chọn nút 'Skip X Configuration' cho phép dừng việc cấu hình X.

P4.11. Chuẩn bị cài đặt

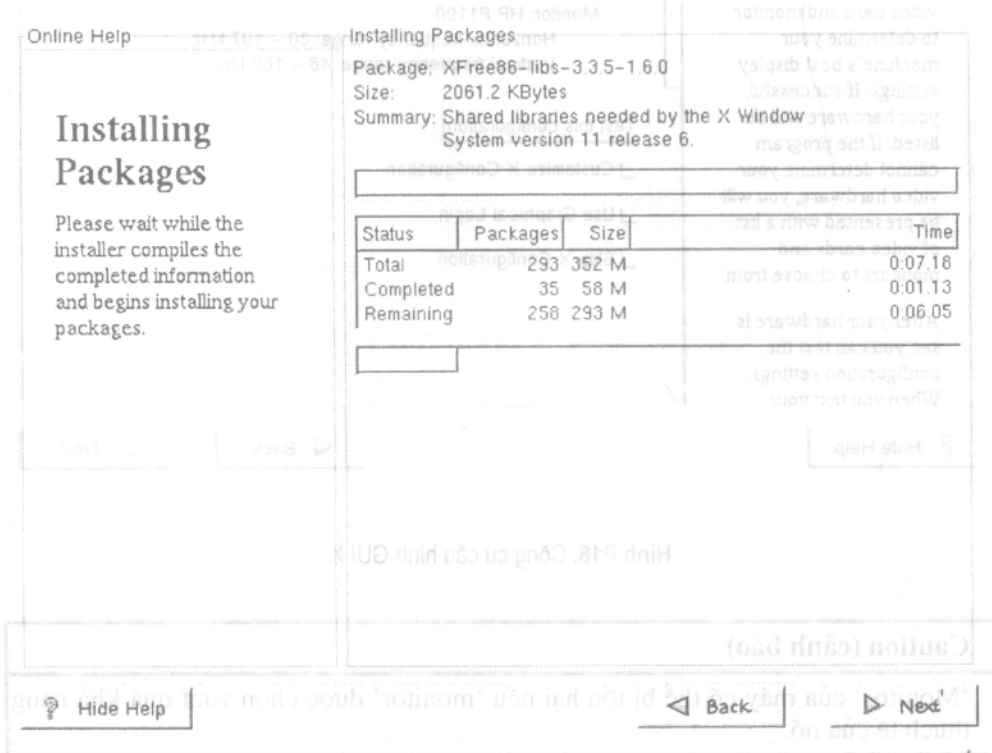
Sau đây là màn hình ở bước sẵn sàng cài đặt Red Hat Linux lên hệ thống.

Warning

Nếu chọn nút 'Next', chương trình cài đặt sẽ viết lên các 'partition' và cài đặt các package. Còn để dừng việc cài đặt trước khi hệ thống của bạn bị thay đổi, hãy khởi động lại hệ thống.

P4.12. Cài đặt các package

Trong bước này, chương trình cài đặt sẽ giữ quyền kiểm soát việc cài đặt tất cả các package được chọn (hình P19), và kết thúc nhanh hay chậm phụ thuộc vào số lượng package được chọn và tốc độ xử lý của máy.

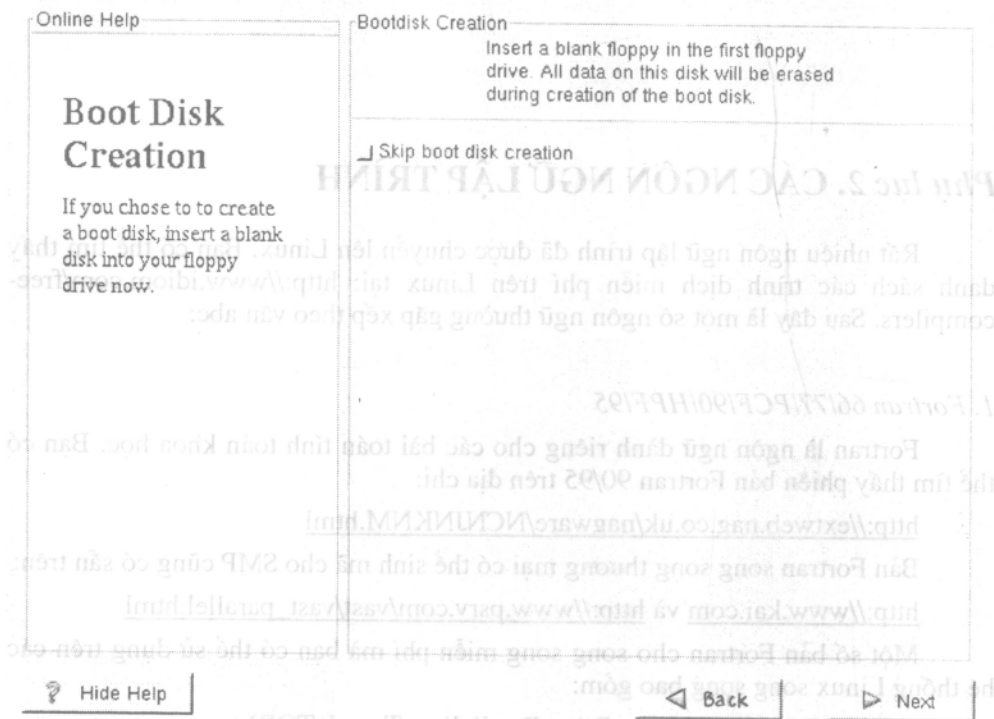


Hình P19. Cài đặt các package

P4.13. Tạo đĩa khởi động

Nếu chọn tạo đĩa khởi động, hãy chèn đĩa trắng vào ổ đĩa mềm (hình P20). Sau đó đĩa khởi động sẽ được tạo. Bạn cũng có thể tạo đĩa khởi động sau khi kết thúc cài đặt, để biết thêm chi tiết, hãy gõ 'man mkbootdisk' ở 'shell prompt' để xem cách hướng dẫn việc tạo đĩa khởi động. Nên tạo đĩa khởi động (được dùng để

khởi động thay LILO) tương ứng với hệ điều hành Red Hat Linux cụ thể trên hệ thống của bạn.



Hình P20. Tạo đĩa khởi động

P4.14. Kết thúc cài đặt

Khi việc cài đặt đã hoàn thành, chương trình cài đặt sẽ chuẩn bị để khởi động lại máy tính của bạn. Nếu bạn đã không cài đặt LILO, việc khởi động lại sẽ thực hiện bằng đĩa khởi động. Sau đó, bạn sẽ thấy 'standard prompt' (dấu mời chuẩn) của LILO như sau: 'boot:'. Tại 'boot: prompt', bạn có thể thực hiện các thao tác sau:

- Ấn 'Enter': cho phép khởi động 'phần tử khởi động mặc định' (default boot entry) của LILO.
- Vào 'Boot Label', rồi ấn 'Enter': LILO sẽ khởi động hệ điều hành (operating system) tương ứng với 'boot label' được đưa vào.
- Không làm gì: sau một khoảng thời gian (mặc định là 5s), LILO sẽ khởi động 'phần tử khởi động mặc định' của nó.

Phụ lục 2. CÁC NGÔN NGỮ LẬP TRÌNH

Rất nhiều ngôn ngữ lập trình đã được chuyển lên Linux. Bạn có thể tìm thấy danh sách các trình dịch miễn phí trên Linux tại: <http://www.idiom.com/free-compilers>. Sau đây là một số ngôn ngữ thường gặp xếp theo vần abc:

1. Fortran 66/77/PCF/90/HPF/95

Fortran là ngôn ngữ dành riêng cho các bài toán tính toán khoa học. Bạn có thể tìm thấy phiên bản Fortran 90/95 trên địa chỉ:

<http://extweb.nag.co.uk/nagware/NCNJNKNM.html>

Bản Fortran song song thương mại có thể sinh mã cho SMP cũng có sẵn trên:

<http://www.kai.com> và http://www.psr.com/vast/vast_parallel.html

Một số bản Fortran cho song song miễn phí mà bạn có thể sử dụng trên các hệ thống Linux song song bao gồm:

- ADAPTOR (Automatic DATA Parallelism TranslaTOR)

<http://www.gmd.de/SCAI/lab/adaptor/adaptor_home.html>

Có thể dịch từ mã HPF (High-Performance Fortran, <<http://www.crpc.rice.edu/HPFF/home.html>>) sang mã Fortran 77/90 với các lời gọi MPI hay PVM.

- Fx <<http://www.cs.cmu.edu/~fx/Fx>>

- HPFC (prototype HPF Compiler,

<<http://www.cri.enscm.fr/~coelho/hpfc.html>>) cho Fortran 77

- PARADIGM (PARAllelizing compiler for DIstributed-memory

General-purpose Multicomputers,

<<http://www.crhc.uiuc.edu/Paradigm/>>)

- Trình dịch Polaris,

<<http://ece.www.ecn.purdue.edu/~eigenman/polaris/>>, được sử dụng trong các cluster PAPERS Linux.

- PREPARE,

<<http://www.irisa.fr/EXTERNE/projet/pampa/PREPARE/prepare.html>>.

- SHPF (Subset High Performance Fortran), kết hợp giữa ADAPT and ADLIB,

<<http://www.ccg.ecs.soton.ac.uk/Projects/shpf/shpf.html>>)

- SUIF (Stanford University Intermediate Form), <<http://suif.stanford.edu/>>

2. *GLU (Granular Lucid)*

GLU (Granular Lucid) là một ngôn ngữ lập trình bậc cao hỗ trợ cả PVM và TCP socket. Các thông tin thêm có thể xem trong:

<<http://www.csl.sri.com/GLU.html>>.

3. *Jade và SAM*

Jade là một ngôn ngữ lập trình song song mở rộng của C cho các clusters sử dụng PVM. Các thông tin thêm có thể xem trong:

<<http://suif.stanford.edu/~scales/sam.html>>.

4. *Mentat và Legion*

Mentat là một ngôn ngữ hướng song song đối tượng dựa trên C++ sử dụng trên các cluster trên Linux. Các thông tin thêm có thể xem trong:

<<http://www.cs.virginia.edu/~mentat/>>.

Ngôn ngữ Legion <<http://www.cs.virginia.edu/~legion/>> được xây dựng trên đỉnh của Mentat, cung cấp máy ảo trên các mạng diện rộng (WAN).

5. *Parallaxis-III*

Parallaxis-III là một ngôn ngữ lập trình cấu trúc được mở rộng từ Modula-2 cho song song hoá dữ liệu (theo mô hình SIMD). Các thông tin thêm có thể xem trong:

<<http://www.informatik.uni-stuttgart.de/ipvr/bv/p3/p3.html>>.

6. *pC++/Sage++*

<<http://www.extreme.indiana.edu/sage/>>.

Phụ lục 3. CÁC NGUỒN THÔNG TIN VỀ LINUX

Hiện nay có rất nhiều tài liệu về hệ điều hành Linux. Để khỏi bị "ngập" thông tin, chúng ta hãy chia các tài liệu theo ba loại người dùng: người mới sử dụng Linux, người sử dụng đã có một số kinh nghiệm nhất định về Linux và người sử dụng cao cấp.

1. Tài liệu cho người mới sử dụng Linux:

- *Linux for Dummies*, by John "maddog" Hall, published by IDG.
- *Using Linux*, by William H. Ball, published by Que;
- *Running Linux*, by Matt Welsh and Lar Kaufman, published by O'Reilly & Associates;
- *Red Hat Linux Secrets*, by Naba Barkakati, published by IDG.
- <http://www.redhat.com>
- <http://www.deja.com>

Bạn có thể tìm thấy các tài liệu của các nhóm Linux trên địa chỉ này:

- linux.help
- linux.redhat
- linux.redhat.digest
- linux.redhat.misc
- linux.redhat.rpm

2. Tài liệu cho người sử dụng đã có một số kinh nghiệm nhất định về Linux :

Bạn có thể tìm thấy nhiều tài liệu hữu ích trong các HOWTO tại thư mục: `/usr/doc/HOWTO`.

3. Tài liệu cho người sử dụng cao cấp :

Cách tốt nhất để bạn có thể hiểu rõ hơn về hệ điều hành Linux là đọc trực tiếp mã nguồn của Linux!

Phụ lục 4. CÁC CÂU HỎI THƯỜNG GẶP (FAQS)

☹ Q (Hỏi): LINUX là gì?

☞ A (Trả lời):

Về phương diện kỹ thuật, LINUX, giống như UNIX, có các chức năng cốt lõi (kernel-based), có chế độ bảo vệ ký ức toàn phần, là một hệ điều hành đa nhiệm do tác giả Linus Torvalds thuộc trường Đại viết Học Helsinki ra vào năm 1991, và phát hành theo bản quyền GNU General Public License.

Linux là một HĐH gọn nhẹ, rất mạnh, miễn phí có thể chạy trên nhiều phần cứng, từ các máy PC, Mac, cho đến Amigas, Alpha ... Linux còn là một công trình tập thể: hàng ngàn nhà phát triển trên khắp thế giới đã liên tục đóng góp các đặc tính, các cách chữa lỗi, để hoàn thiện nó.

Ngoài việc là một hệ HĐH miễn phí, LINUX còn là một phần mềm có mã nguồn mở: có nghĩa là chẳng những bạn dùng các chương trình có sẵn để chạy trên máy của mình, bạn còn có thể sửa chữa, phát triển từ bộ mã nguồn theo sở thích của bạn.

Linux đang ở thời kỳ phát triển, theo công ty Red Hat ước lượng, có khoảng 7 triệu người dùng LINUX trên khắp thế giới, và có 750.000 server đang dùng LINUX.

☹ Q: Liệu cái máy tính của tôi có chạy được LINUX không?

☞ A:

Có lẽ đặc điểm vượt trội hơn hết của LINUX chính là nó chạy được trên hầu hết loại máy tính cơ bản.

Máy desktop: Bao gồm các phần cứng của Intel và Intel tương thích, các chip PowerPC, Sun Sparcs, DEC Alpha và nhiều loại khác. Bạn cũng không cần thiết phải có đời máy hiện đại nhất, mới nhất, LINUX có tính môđun cao, nếu loại bớt các thứ "phụ tùng", LINUX có thể chạy trên máy 386 và chỉ cần 150 MB đĩa trống, 2MB RAM. Một số nhà phát triển còn viết LINUX chạy từ một đĩa mềm duy nhất !

Máy laptop: Linux cũng hoạt động tốt trên nhiều laptop như Apple PowerBooks, IBM ThinkPad, Toshiba Tecras.

Điều kỳ diệu là LINUX có thể chạy cả trên Intel và Mac, ít nhất là với các phần cứng cơ bản (bo mạch chính, bộ vi xử lý, bộ nhớ) Vấn đề có thể làm bạn đau khổ nhất đó chính là các thiết bị ngoại vi.

☹ **Q: Liệu các thiết bị ngoại vi của tôi có hoạt động được với LINUX không?**

☞ **A:**

Được và không ! Các thiết bị căn bản như modem, máy in, network adapter hoạt động với LINUX. Có thứ làm việc tốt, có thứ không tốt lắm, và có thứ hoàn toàn không. Sau đây là các nét chung:

- Các card ISA cũ: Nhiều thứ hoạt động rất tốt với LINUX như Sound BLaster 16, US
- Robotic Sportster modem. Dùng được các món đồ cổ này chính là một ưu thế của LINUX.
- Card PCI: Về nguyên tắc LINUX dung nạp card ISA tốt hơn PCI, nhiều modem PCI là Windows modem, nên chúng không thể làm việc với LINUX. Các sound card PCI đời mới như Turtle Beach Montego và Sound Blaster Live không hoạt động được với LINUX. Nhiều card Ethernet PCI là SCSI thì lại được LINUX hỗ trợ. Vì thế bạn nên kiểm tra trước tại các nhà phân phối danh sách các thiết bị được LINUX hỗ trợ.
- Cắm là chạy (plug and play): Quả thật nó làm cho cuộc sống với Windows của chúng ta tươi đẹp hơn. LINUX cũng làm được việc này, có điều đôi khi không mấy suông sẻ. Tuy thiết bị, có thể bạn chẳng gặp chút rắc rối nào, nhưng cũng có khi bạn cũng phải toát mồ hôi.
- Ngoại vi cho Windows: Nhằm hạ giá thành, một số phần cứng được thiết kế chỉ dùng riêng cho Windows mà thôi, như modem và printer - vì thế nó sẽ không làm việc với LINUX. Lý do là vì các thiết bị ngoại vi này (ví dụ như 3Com/U.S. Robotics Winmodem và Lexmark Winwriter 200 printer cộng sinh với các phần mềm và CPU của máy tính để hoạt động.
- USB: đang phát triển mạnh, nhưng rủi thay LINUX chưa đáp ứng kịp. Nhưng ít nhất cũng đang có một nhà phát triển viết chương trình cho LINUX dùng bus này.

Tóm lại, bạn nên hỏi các nhà phân phối về danh sách các thiết bị được LINUX hỗ trợ.

☹ **Q: Làm thế nào để tải xuống chương trình LINUX ?**

☞ **A:**

Lời khuyên đầu tiên dành cho bạn: Chớ nên tải xuống. Hãy mua một bản copy. Với giá dưới 50 USD bạn có trọn bộ - đĩa CD-ROM kèm với các quyển sách

kinh điển về LINUX. Các nhà phân phối LINUX có tiếng là Red Hat, Caldera, Debian. Mỗi nhà phát hành đều có ưu, khuyết điểm riêng: Red Hat khá phổ biến vì giao diện cài đặt dễ hiểu, Debian thì có đủ thứ đi kèm với LINUX dành cho bạn, còn thế mạnh của Caldera là hỗ trợ mạng.

⊗ **Q: Làm thế nào để truy cập Internet với LINUX?**

☞ **A:**

Truy cập Internet bằng LINUX sẽ gian nan hơn với Windows. Thay vì chỉ click theo các tiện ích hướng dẫn truy cập Internet kiểu Windows, LINUX đòi hỏi bạn phải hiểu nhiều về các thông tin truy cập. Nhưng may mắn là bạn cũng có các chương trình tiện ích giúp sức.

Trước tiên phải cài đặt các giao thức, tiện ích và module để truy cập khi cài đặt LINUX.

Sau đó bạn xác lập nối kết như PPP, ISP. Có các tiện ích dùng giao diện đồ hoạ giúp bạn thực hiện dễ dàng hơn, giống như Dial-Up Network trong Windows, bạn chỉ nhập ID login, password số điện thoại của nhà ISP, địa chỉ DNS ... Chương trình sẽ lo các việc còn lại cho bạn.

Thông thường các bản phát hành của LINUX có chứa đủ các tiện ích để bạn truy cập Internet. Bạn sẽ tìm thấy các giao thức, Netscape Navigator, Lynx, chương trình e-mail.

⊗ **Q: Làm thế nào để tôi có thể chạy LINUX cùng với Windows?**

☞ **A:**

Bạn hoàn toàn có thể dùng LINUX đồng thời chia sẻ máy tính với những ai còn muốn sử dụng Windows, vì bạn có thể chạy hai hệ điều hành này trên cùng một máy tính. Bạn phải dùng kiểu "khởi động chọn lựa", chạy Windows hay LINUX ngay khi boot máy. (Chương trình LILO kèm theo Red Hat LINUX)

Còn nếu thỉnh thoảng mới chạy Windows, bạn có thể chọn Wine. Tiện ích miễn phí này cho phép nhiều chương trình ứng dụng Windows chạy dưới LINUX. Trang chủ của Wine cung cấp các thông tin cần thiết về các ứng dụng nó có hỗ trợ, bạn có thể tham khảo trước khi quyết định.

Còn một tùy chọn khác: VMware for LINUX cho phép chạy Windows 3.1, 95, 98, NT 4.0 trong LINUX. Bạn cần Pentium với tối thiểu 64 MB RAM (VMware đề nghị Pentium II, 96MB RAM). Đây là giải pháp lý tưởng cho những ai yêu thích LINUX nhưng chưa đủ can đảm để đoạn tuyệt với Windows.

⊗ **Q: Làm thế nào để tôi có thể chạy Web site của tôi trên LINUX?**

☞ **A:**

Internet cung cấp hầu hết động lực cho LINUX có được ngày hôm nay, cho nên không có gì đáng ngạc nhiên khi có vô số công cụ giúp các bạn soạn và chạy

một Web site chạy trên LINUX. Thực tế đang có rất nhiều ISP dùng LINUX chạy trên server của họ. Nếu bạn định thiết kế một trang Web chạy trên LINUX, hãy tìm các trang chủ có server chạy LINUX như CI Host và Web Serve Pro.

Chọn một Web server: Một việc quan trọng khác ngoài HĐH, đó là Web server. Thường chương trình Apache là phổ biến nhất. Đây là một chương trình server rất mạnh mà các công ty về Web sử dụng để tải các trang Web.

⊗ **Q: Làm thế nào tôi có thể sử dụng đĩa mềm trên Red Hat Linux?**

☞ **A:**

- Sử dụng *Mtools*

Nếu bạn có một đĩa mềm theo khuôn dạng MS-DOS, bạn có thể sao nó bằng công cụ *mtools*. Ví dụ, để copy file *vd.txt* từ đĩa mềm MS-DOS, bạn sử dụng lệnh:

```
$ mcopy a:vd.txt vd.txt
```

Để thay đổi thư mục hiện thời trên đĩa mềm, chẳng hạn *subdir*, bạn sử dụng lệnh:

```
$ mcd a:subdir
```

- Hệ thống file ext2 (Second Extended)

Hệ thống file ext2 là hệ thống file được Linux sử dụng. Ví dụ để tạo một đĩa mềm sử dụng hệ thống file ext2, bạn sử dụng lệnh:

```
$ /sbin/mke2fs /dev/fd0
```

Trên Linux, thư mục */dev/fd0* tương ứng với ổ đĩa mềm thứ nhất.

⊗ **Q: Cách tắt máy (Shut Down) đúng trên Linux?**

☞ **A:**

Bạn đừng bao giờ tắt công tắc máy khi chưa được phép. Sau đây là cách tắt máy đúng:

- Từ X Window: Chọn *Log out* từ Menu chính, sau đó bạn chọn *Halt* hoặc *Reboot*. Bạn sẽ phải gõ mật khẩu của bạn để có thể dừng hệ thống.
- Từ dấu nhắc của shell: Bạn gõ lệnh *shutdown -r* hoặc *shutdown -h now*.
- Từ màn hình *login*: Bạn vào mục *Option* trong *System*. Bạn có thể chọn *halt* (dừng hệ thống) hoặc *reboot* (khởi động lại hệ thống), cách này không yêu cầu mật khẩu.

Phím tắt để dừng hệ thống: Bạn có thể dùng tổ hợp phím Ctrl-Alt-Del để dừng hệ thống tương tự như trên Windows.

⊗ Q: Cách dùng lại các lệnh đã sử dụng trên Linux?

A:

Tìm kiếm trong file *.bash_history*:

File này lưu trữ 500 lệnh gần đây nhất mà bạn đã sử dụng từ dòng lệnh của shell. Bạn có thể xem file này bằng lệnh *less* chẳng hạn:

```
$ less .bash_history
```

Sau đó bạn dùng phím space để lật trang, phím B để trở lại trang trước hoặc Q để thoát.

Tuy nhiên tìm từng trang như vậy là một công việc rất mất thời gian. Một cách khác là bạn có thể tìm kiếm bằng lệnh *history* kết hợp với *grep*, ví dụ để tìm kiếm lệnh *man*, bạn dùng lệnh:

```
$ history | grep man
```

Sau đây là một số lệnh tắt để sử dụng các lệnh đã sử dụng:

```
$ !!
```

(gọi là "bang bang") sẽ thực hiện lệnh gần đây nhất đã sử dụng.

```
$ !<number>
```

(ví dụ \$!302) sẽ thực hiện lệnh số 302 trong file *.bash_history*.

```
$ !<string>
```

(ví dụ \$!rpm) sẽ thực hiện lệnh chứa rpm trong file *.bash_history*.

Bạn cũng có thể chỉ cần dùng các phím lên xuống để duyệt qua các lệnh đã sử dụng cho đến khi bạn tìm được lệnh cần dùng, khi đó bạn chỉ cần gõ Enter để thực hiện lệnh.

⊗ Q: Máy tính của tôi có 128 MB RAM, nhưng Linux chỉ nhận được 64 MB. Làm thế nào để tôi có thể sử dụng được toàn bộ RAM tôi có?

A:

Thông thường, hạt nhân Linux sẽ tự động nhận biết RAM trong máy tính của bạn. Tuy nhiên trong trường hợp này, bạn có thể tự đặt dung lượng RAM cho máy tính của bạn.

Trước hết, bạn hãy kiểm tra dung lượng RAM đang được sử dụng, ví dụ:

```
$ cat /proc/meminfo
```

```
total:      used:      free:      shared: buffers: cached:
```

```
Mem:  64655360 63033344 1622016 51159040 1552384 33816576
```

```
Swap: 73990144 700416 73289728
```

```
MemTotal:      63140 kB
```

```
MemFree:      1584 kB
MemShared:    49960 kB
Buffers:      1516 kB
Cached:       33024 kB
SwapTotal:    72256 kB
SwapFree:     71572 kB
```

Bây giờ để sử dụng hết 128 MB RAM, bạn có thể gõ trung tiếp từ dấu mời *boot:* của chương trình LILO:

```
boot: linux mem=128M
```

Nếu bạn không muốn gõ mỗi lần khởi động như vậy, bạn có thể sửa đổi lựa chọn trong file `/etc/lilo.conf`. Để làm điều này, bạn phải là *root*, sau đó bạn sử dụng một hệ soạn thảo, chẳng hạn *vi*, để sửa file `/etc/lilo.conf`. Ví dụ:

```
[root@localhost root]# vi /etc/lilo.conf
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
image=/boot/vmlinuz-2.2.5-15
        label=linux
        root=/dev/hda1
        read-only
```

Trong file này, bạn thêm vào đầu file dòng lệnh: `append="mem=128M"`.

Bây giờ bạn gõ `/sbin/lilo` để khởi động lại LILO. Sau đó khi bạn khởi động lại máy, cấu hình bộ nhớ của bạn sẽ được chuyển cho hạt nhân Linux.

Phụ lục 5. MỘT SỐ NHÀ CUNG CẤP HĐH LINUX PHỔ BIẾN

HĐH Linux được cung cấp bởi rất nhiều distro, khi đó HĐH Linux mang những cái tên khác nhau như RedHat, Fedora, Mandrake, SuSe hay Debian. Điểm chung giữa các hệ điều hành này là phần nhân (phần lõi) của HĐH Linux. Sự khác biệt nằm ở các tiện ích, các phần mềm đi cùng. Tùy thuộc vào mục đích nhằm tới đối tượng nào, các distro đóng gói phần nhân Linux với các gói phần mềm tiện ích. Do vậy, việc cài đặt cũng như cấu hình có thể không đồng nhất. Ví dụ, Fedora là thế hệ sau của RedHat, hướng tới những người sử dụng thuần túy. Do vậy, ở mức mặc định, hệ thống cài đặt khá nhiều phần mềm tiện ích cho mọi lĩnh vực. Ngoài ra, hệ thống cũng cung cấp các giao diện đồ họa khá thuận tiện và thân thiện. Ngược lại, Debian lại hướng tới những người dùng chuyên nghiệp. Trong lúc cài đặt, người dùng có thể cấu hình để hệ thống nạp những module cần thiết. Ưu điểm của Debian là tốc độ, tính ổn định và khả năng kiểm soát hệ thống cao. Tuy nhiên, với người dùng thuần túy, sử dụng Debian thực sự là một thách thức rất lớn. Hướng tới sự gọn nhẹ và tiện lợi, người ta còn nhắc đến Knopix với khả năng chạy hệ điều hành trên một đĩa CD.

Trong phụ lục này, chúng tôi cố gắng cập nhật cho bạn đọc những thông tin cô đọng nhất về một số HĐH Linux của những distro có tiếng trên thế giới bao gồm RedHat, Mandrake và Debian. Với các hệ điều hành khác, bạn đọc có thể tham khảo trên mạng Internet thông qua các cái tên đã được chúng tôi nhắc tới ở trên.

RedHat và Fedora

Fedora là tên gọi sau này của các HĐH Linux của RedHat cho máy trạm. Người ta sẽ không thấy các phiên bản RedHat 10, RedHat 11 hay RedHat 12 mà

thay vào đó là Fedora Core 1, Fedora Core 2 và Fedora Core 3. Tuy nhiên, không chỉ đơn giản là thay tên, sự khác biệt giữa Fedora với các HĐH RedHat tiền thân nằm ở 3 điểm lớn sau đây:

- Đây là một dự án hỗ trợ cộng đồng. Điều đó có nghĩa là bản thân người dùng cũng có thể tạo ra một phiên bản mới (một Fedora Core mới, ví dụ Fedora Core 4, 5 ...).
- Các phiên bản mới được cập nhật 6 tháng một lần.
- Khi một phiên bản mới ra đời, nhà cung cấp chỉ có trách nhiệm hỗ trợ phiên bản trước đó trong thời gian từ 6 đến 8 tháng. Trong thời gian này, người dùng có thể lấy các patch sửa lỗi từ dự án Fedora Legacy Project. Tuy nhiên, RedHat khuyến cáo người dùng nên thường xuyên cập nhật HĐH của mình bằng một phiên bản mới sau từ 6 đến 8 tháng, vì việc nâng cấp hệ điều hành bằng một Fedora Core mới tương đối dễ dàng.

Đối với những người dùng HĐH Linux của RedHat cần lưu ý thêm một điểm nữa. Hiện tại, RedHat phát hành song song hai dòng HĐH Linux, dòng sản phẩm hướng tới sử dụng cá nhân là Fedora, dòng sản phẩm hướng tới môi trường doanh nghiệp là RedHat Linux Enterprise (RHLE). Trong khi người dùng có thể cập nhật miễn phí Fedora, các doanh nghiệp phải trả tiền để được sử dụng RHLE. Ngược lại, nếu như RedHat chỉ hỗ trợ người dùng Fedora trong vòng 6 đến 8 tháng kể từ khi có phiên bản mới ra đời, thời gian hỗ trợ đối với RHLE là 5 năm.

Mandrake

Mandrake cũng là một HĐH Linux khác được sử dụng khá phổ biến cho các máy tính cá nhân. Hiện tại, phiên bản mới nhất của Mandrake là 10.2. Ưu điểm của Mandrake là nhẹ, không đòi hỏi nhiều tài nguyên và được hỗ trợ rất nhiều phần mềm, đặc biệt là các phần mềm phục vụ các mục đích nghiên cứu. Do vậy, Mandrake được sử dụng phổ biến trong môi trường phòng thí nghiệm, trường đại học. Trong phiên bản gần nhất, phiên bản 10.1 (đưa ra thị trường vào ngày 13 tháng 2 năm 2005), ta có thể thấy sự có mặt của một số phần mềm tiêu biểu bao gồm:

- Linux Kernel 2.6.8 (and various fixes from 2.6.9rc)
- Xorg 6.7.0
- KDE 3.2.3
- GNOME 2.6
- Glibc 2.3.3, GCC 3.4.1

- Apache 2.0.50, PHP 4.3.8
- MySQL 4.0.18, Samba 3.0.6
- Mozilla 1.7.2, GIMP 2.0.4
- OpenOffice.org 1.1.3

Mặc dù Mandrake có khá đầy đủ các phần mềm cung cấp các dịch vụ mạng cũng như các công cụ để quản lý, nhưng nó không được khuyến cáo để sử dụng như HĐH cho các máy chủ.

Một điểm đặc biệt nữa của Mandrake là khả năng hỗ trợ đa ngôn ngữ rất mạnh. Với người dùng Việt Nam, Mandrake 10.0 không chỉ đơn thuần là một HĐH giao diện tiếng Việt, hơn thế, HĐH này hỗ trợ rất nhiều đặc tính như bộ gõ tiếng Việt, trợ giúp tiếng Việt.

Debian

Debian là một HĐH hành Linux khác được cung cấp bởi dự án Debian. Dự án này được triển khai bởi một hiệp hội các cá nhân mong muốn tạo ra một hệ điều hành miễn phí. Tên đầy đủ của hệ điều hành này là Debian GNU Linux. Tuy nhiên, người ta thường nhắc đến nó bằng một tên khác ngắn gọn hơn Debian. Dự án Debian được khởi sự chính thức vào ngày 16 tháng 8 năm 1993 bởi Ian Mardock. Tác giả của dự án muốn thông qua dự án của mình đưa ra một khái niệm mới về các nhà phân phối mang tính mở hơn. Tuy nhiên, Debian chỉ thực sự mở với những người có tham gia và đóng góp tích cực vào dự án này.

Debian được xem là một ví dụ tiêu biểu về sự tinh giản và hướng tới hiệu năng và sự ổn định. Với nguyên tắc lược bỏ những gì không cần thiết, Debian đem lại cho người dùng hiệu năng cao hơn bất cứ HĐH nào. Tuy nhiên, không phải vì thế Debian nghèo nàn về phần mềm, ngược lại, Debian có một kho phần mềm vô cùng phong phú cho phép người dùng cập nhật theo nhiều cách thức khác nhau. Debian cung cấp tới 8710 gói phần mềm đã biên dịch theo một khuôn dạng *.deb. Với các gói tin dạng này, người dùng dễ dàng cài đặt bằng một lệnh duy nhất. Tất cả các phần mềm này đều được cung cấp miễn phí. Lưu ý rằng, các gói phần mềm theo định dạng của RedHat (*.rpm) không thể cài đặt trực tiếp trên các hệ thống Debian. Tuy nhiên, điều đó không có nghĩa là các gói *.rpm không thể sử dụng được trên hệ điều hành Linux Debian. Debian cung cấp một phần mềm tên là alien cho phép chuyển một gói phần mềm ở dạng *.rpm sang *.deb. Debian còn có một điểm nổi bật giúp hệ thống ổn định trong các tình huống cài đặt và cập nhật. Đặc tính này được gọi là "đóng gói vi mô" (micro packaged). Cụ thể, đó là việc sử dụng

các thông tin về sự phụ thuộc giữa các gói tin để đảm bảo tính nhất quán của hệ thống.

TÀI LIỆU THAM KHẢO

1. Sách

- [1]. Matt Welsh, Phil Hughes, *Linux Installation and Getting Started*, 1996.
- [2]. J. Tackett, D. Gunter, L. Brown, *Special Edition Using Linux*, Que Corporation, 1995.
- [3]. Matt Welsh, Lar Kaufman, *Running Linux*, O'Reilly Associated Inc., 1996.
- [4]. E. Nemeth et al., *Unix System Administration Handbook*, Prentice Hall Inc., 1989.
- [5]. Mark G. Solbell, *A Pratical Guide To The Unix System*, The Benjamin/Cummings Publishing Company, 1994.
- [6]. M. Beck et al., *Linux Kernel Internal*, Addison Wesley Press, 1998.

2. Các địa chỉ Internet

- [7]. www.linuxdocs.org - Các tài liệu Linux Howto.
- [8]. www.linux.org - Các thông tin liên quan đến Linux.
- [9]. www.gnu.org - Tài liệu và các chương trình Linux.

NGUYỄN THANH THỦY (Chủ biên),
NGUYỄN QUANG HUY, NGUYỄN HỮU ĐỨC, ĐINH LAN ANH

Nhập môn

HỆ ĐIỀU HÀNH LINUX

Chịu trách nhiệm xuất bản: PGS, TS TÔ ĐĂNG HẢI

Biên tập: ĐẶNG VĂN SỬ

NGUYỄN NGỌC KHUÊ

Trình bày bìa: TRẦN THẮNG

NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

70 Trần Hưng Đạo - Hà Nội

In 800 cuốn khổ 16 x 24 cm, tại Công ty in Hàng không.

Giấy phép xuất bản số : 546 – 36 – 21 / 4 / 2005..

In xong và nộp lưu chiểu tháng 10/2005.

205214



Giá: 26.000^d