Linux: Instalação e Iniciação

Matt Welsh
Phil Hughes
David Bandel
Boris Beletsky
Sean Dreilinger
Robert Kiesling
Henry Pierce
Gleydson Mazioli da Silva

Tradução Oficial Alfamídia Fernando Miguel de Alava Soto Rodrigo de Losina Silva Alexandre Folle de Menezes

Versão 1.3, julho/2002.

Os Capítulos 3, 5 e 8 são adaptações do texto:

Guia Foca GNU/Linux

Copyleft © 1999-2002 - Gleydson Mazioli da Silva.

Fonte: http://focalinux.cipsga.org.br

O restante do texto é uma adaptação do original: <u>Linux Installation and Getting</u> Started

Copyright ©1992-1996 Matt Welsh

Copyright ©1998 Specialized Systems Consultants, Inc (SSC)

P.O. Box 55549

Seattle, WA 98155-0549

USA

Phone: +1-206-782-7733 Fax: +1-206-782-7191 E-mail: ligs@ssc.com

URL: http://www.ssc.com/

Tradução autorizada pelo autor.

As marcas registradas utilizadas no decorrer deste documento são usadas unicamente para fins didáticos, sendo estas propriedade de suas respectivas companhias.

A Alfamídia não assume qualquer responsabilidade por erros ou omissões, ou por danos resultantes do uso das informações contidas neste livro.

Tradução para português - Brasil, 2002:

Alfamídia Ltda

Rua Félix da Cunha, 818 Porto Alegre - RS - Brasil Fone/Fax: +55 (51) 3346-7300

E-mail: alfamidia@alfamidia.com.br URL: http://www.alfamidia.com.br

Tradução para português:

Fernando Miguel de Alava Soto (soto@alfamidia.com.br)

Rodrigo de Losina Silva (rodrigo@alfamidia.com.br)

Alexandre Folle de Menezes (menezes@alfamidia.com.br)

Permissão para copiar, distribuir, alterar, conforme licença GNU, desde que sejam mantidos os créditos acima, incluindo nome e dados da empresa e nome dos tradutores.

NOTA DO TRADUTOR: FORAM TRADUZIDOS OS CAPÍTULOS 1 a 3 do Guia "Linux Installation and Getting Started", de Matt Welsh. O sinal (.....) marca onde partes do texto foram suprimidas na tradução. Basicamente, foram suprimidos textos considerados pelo tradutor como partes que necessitam uma atualização mais profunda ou partes que fazem referência aos demais capítulos ou partes não traduzidas.

Linux Installation and Getting Started is a free document; you may reproduce and/or modify it under the terms of version 2 (or, at your option, any later version) of the GNU General Public License as published by the Free Software Foundation.

This book is distributed in the hope it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details, in Appendix C.

The authors encourage wide distribution of this book for personal or commercial use, provided the above copyright notice remains intact and the method adheres to the provisions of the GNU General Public License (see Appendix C). In summary, you may copy and distribute this book free of charge or for a profit. No explicit permission is required from the author for reproduction of this book in any medium, physical or electronic.

Note, derivative works and translations of this document *must* be placed under the GNU General Public License, and the original copyright notice must remain intact. If you have contributed new material to this book, you must make the source code (e.g., LaTeX source) available for your revisions. Please make revisions and updates available directly to the document maintainers, Specialized Systems Consultants. This will allow for the merging of updates and provide consistent revisions to the Linux community.

If you plan to publish and distribute this book commercially, donations, royalties, and/or printed copies are greatly appreciated by the authors and the Linux Documentation Project. Contributing in this way shows your support for free software and the Linux Documentation Project. If you have questions or comments, please contact SSC.

CONTEÚDO

CONTEÚDO4		
1. I	NTRODUÇÃO	11
1.1	SOBRE ESTE LIVRO	11
1.2	UMA BREVE HISTÓRIA DO LINUX	12
1.3	RECURSOS DO SISTEMA	14
1.4	RECURSOS DE SOFTWARE	15
1.4.1	FERRAMENTAS E LINGUAGENS DE PROGRAMAÇÃO	16
1.4.2	3	
1.4.3	Introdução às Redes	18
1.4.4	TELECOMUNICAÇÕES E SOFTWARES DE BBS	19
1.4.5	WORLD WIDE WEB	19
1.4.6	INTERFACE COM O MS-DOS	19
1.4.7	OUTRAS FERRAMENTAS	20
1.5	QUESTÕES DE COPYRIGHT	20
1.6	O PROJETO E A FILOSOFIA DO LINUX	
1.7	DIFERENÇAS ENTRE O LINUX E OUTROS SISTEMAS OPERACIONAIS	
1.7.1		
1.7.2		
1.7.3	LINUX VS OUTRAS IMPLE MENTAÇÕES DE UNIX	24
1.8	EXIGÊNCIAS DE HARDWARE	
1.8.1	PLACA-MÃE E CPU	24
1.8.1 1.8.2	PLACA-MÃE E CPU Exigências de Memória	24
1.8.1 1.8.2 1.8.3	Placa-mãe e CPU Exigências de Memória Controladores de Disco Rígido	24 25 25
1.8.1 1.8.2 1.8.3 1.8.4	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO	24 25 25
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX DOCUMENTAÇÃO ONLINE	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2	PLACA-MÃE E CPU. EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO. ESPAÇO EM DISCO. OUTROS DISPOSITIVOS. FONTES DE INFORMAÇÃO SOBRE O LINUX. DOCUMENTAÇÃO ONLINE. LINUX NA WEB.	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX DOCUMENTAÇÃO ONLINE LINUX NA WEB LIVROS E OUTROS TRABA LHOS PUBLICADOS	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4	PLACA-MÃE E CPU. EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO. ESPAÇO EM DISCO. OUTROS DISPOSITIVOS. FONTES DE INFORMAÇÃO SOBRE O LINUX. DOCUMENTAÇÃO ONLINE. LINUX NA WEB. LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET.	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4 1.9.5	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX DOCUMENTAÇÃO ONLINE LINUX NA WEB LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET LISTAS DE E-MAIL POR INTERNET	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4	PLACA-MÃE E CPU. EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO. ESPAÇO EM DISCO. OUTROS DISPOSITIVOS. FONTES DE INFORMAÇÃO SOBRE O LINUX. DOCUMENTAÇÃO ONLINE. LINUX NA WEB. LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET.	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4 1.9.5 1.10	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX DOCUMENTAÇÃO ONLINE LINUX NA WEB LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET LISTAS DE E-MAIL POR INTERNET	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4 1.9.5 1.10	PLACA-MÃE E CPU. EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO. ESPAÇO EM DISCO. OUTROS DISPOSITIVOS. FONTES DE INFORMAÇÃO SOBRE O LINUX. DOCUMENTAÇÃO ONLINE. LINUX NA WEB. LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET. LISTAS DE E-MAIL POR INTERNET. BUSCANDO AJUDA SOBRE LINUX.	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4 1.9.5 1.10	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX DOCUMENTAÇÃO ONLINE LINUX NA WEB LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET LISTAS DE E-MAIL POR INTERNET BUSCANDO AJUDA SOBRE LINUX	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4 1.9.5 1.10 2 7 2.1	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX DOCUMENTAÇÃO ONLINE LINUX NA WEB LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET LISTAS DE E-MAIL POR INTERNET BUSCANDO AJUDA SOBRE LINUX INTRODUÇÃO CONCEITOS BÁSICOS DO LINUX	
1.8.1 1.8.2 1.8.3 1.8.4 1.8.5 1.9 1.9.1 1.9.2 1.9.3 1.9.4 1.9.5 1.10 2 1 2.1 2.2	PLACA-MÃE E CPU EXIGÊNCIAS DE MEMÓRIA CONTROLADORES DE DISCO RÍGIDO ESPAÇO EM DISCO OUTROS DISPOSITIVOS FONTES DE INFORMAÇÃO SOBRE O LINUX DOCUMENTAÇÃO ONLINE LINUX NA WEB LIVROS E OUTROS TRABA LHOS PUBLICADOS GRUPOS DE DISCUSSÃO DA USENET LISTAS DE E-MAIL POR INTERNET BUSCANDO AJUDA SOBRE LINUX FUTORIAL SOBRE LINUX INTRODUÇÃO CONCEITOS BÁSICOS DO LINUX CRIANDO UMA CONTA	

2.2.4	SHELLS E COMANDOS	32
2.2.5	LOGOUT	33
2.2.6	Mudando sua senha	33
2.2.7	ARQUIVOS E DIRETÓRIOS	33
2.2.8	A ÁRVORE DE DIRETÓRIO S	34
2.2.9	O DIRETÓRIO DE TRABALHO CORRENTE	35
2.2.10	O REFERINDO-SE A DIRETÓRIOS HOME	36
2.3	PRIMEIROS PASSOS NO LINUX	36
2.3.1	Passeando por aí	36
2.3.2	VISUALIZANDO O CONTEÚ DO DE DIRETÓRIOS	37
2.3.3	CRIANDO NOVOS DIRETÓR IOS	39
2.3.4	COPIANDO ARQUIVOS	39
2.3.5	MOVENDO ARQUIVOS	40
2.3.6	EXCLUINDO ARQUIVOS E DIRETÓRIOS	40
2.3.7	VISUALIZANDO ARQUIVOS	40
2.3.8	BUSCANDO AJUDA ONLINE	41
2.4	COMANDOS BÁSICOS DO UNIX	42
2 0	OMANDOS DADA MANIDULAÇÃO DE DIDETÓDIOS	42
	OMANDOS PARA MANIPULAÇÃO DE DIRETÓRIOS	
	O COMANDO LS	
	O COMANDO CD	
	O COMANDO PWD	
	O COMANDO MKDIR	
3.5	O COMANDO RMDIR	45
4 E	XPLORANDO O SISTEMA DE ARQUIVOS	46
	OMANDOS PARA MANIPULAÇÃO DE ARQUIVOS	
5.1	O COMANDO RM	50
5.2	O COMANDO CP	50
5.3	O COMANDO MV	51
5.4	O COMANDO LN	52
5.5	O COMANDO CAT	53
5.6	O COMANDO TAC	53
5.7	O COMANDO MORE	54
5.8	O COMANDO LESS	54
	O COMANDO HEAD	
5.10	O COMANDO TAIL	
5.11	O COMANDO TOUCH	
5.11	O COMANDO WC	
5.13	O COMANDO SORT	
5.14	O COMANDO DIFF	58

6 I	PERMISSÕES DE ARQUIVOS	60
6.1	CONCEITOS DE PERMISSÕ ES DE ARQUIVOS	60
6.2	INTERPRETANDO PERMISS ÕES DE ARQUIVOS	60
6.3	DEPENDÊNCIAS DE PERMISSÕES	61
6.4	MODIFICANDO PERMISSÕE S	62
7 (GERENCIANDO LINKS DE ARQUIVOS	63
7.1	LINKS RÍGIDOS	63
7.2	LINKS SIMBÓLICOS	64
8 (COMANDOS DE BUSCA	65
8.1	O COMANDO GREP	65
8.2	O COMANDO FIND	65
8.3	O COMANDO WHICH	67
8.4	O COMANDO WHEREIS	68
9 U	USANDO O EDITOR VI	69
9.1	CONCEITOS.	69
9.2	INICIANDO O VI	70
9.3	INSERINDO TEXTO	70
9.4	APAGANDO TEXTO	72
9.5	MODIFICANDO TEXTO	73
9.6	COMANDOS PARA MOVIMEN TAR O CURSOR	74
9.7	SALVANDO ARQUIVOS E S AINDO DO VI	75
9.8	EDITANDO OUTRO ARQUIVO	75
9.9	INCLUINDO OUTROS ARQUIVOS	76
9.10	RODANDO COMANDOS DO S HELL	77
9.11	AJUDA NO VI	77
10 A	ACESSANDO ARQUIVOS MS-DOS	78
11 (O SHELL	79
11.1	TIPOS DE SHELL	79
11.2	CARACTERES CORINGA	80
11.3	REDIRECIONAMENTOS E P IPES	82
11.3	.1 ENTRADA PADRÃO E SAÍD A PADRÃO	82
11.3	2.2 REDIRECIONANDO ENTRAD A E SAÍDA	83
11.3	3.3 REDIRECIONAMENTO DE S AÍDA NÃO-DESTRUTIVO	84
11.3	.4 USANDO PIPES	85
11.4	PERSONALIZANDO SEU AM BIENTE	86
11.4	1 VARIÁVEIS DO SHELL E O AMBIENTE	86
11.4	1.1 A variável de ambiente PATH	88
11.4	- 2 SCRIPTS DE INICIALIZA CÃO DO SHELL	89

12 C	ONTROLE DE TAREFAS	91
12.1	TAREFAS E PROCESSOS	91
12.2	FOREGROUND AND BACKGR OUND	92
12.3	COLOCANDO TAREFAS EM SEGUNDO PLANO E ABOR TANDO TAREFAS	92
12.4	PARANDO REINICIANDO T AREFAS	94
13 O	BTENDO E INSTALANDO O LINUX	97
13.1	INSTALAÇÃO GENÉRICA	97
13.1.1	Principais distribuições Linux	97
13.1.2	PLANEJAMENTO BÁSICO	98
13.1.3	HARDWARE	98
13.1.4	PLANEJAMENTO	99
13.1.5	PLANILHA PARA PLANEJA MENTO DO SISTEMA	99
13.1.6	Mouse	101
13.1.7	DISCOS RÍGIDOS (HDS) E CD-ROMS	101
13.1.8	DISCOS NO LINUX	102
13.1.9	Instalando o X Window	102
13.1.1	0 HARDWARE DE REDE	103
13.1.1	1 Planejamento, Parte 2	103
13.1.1	2 ESTRATÉGIAS DE PARTIC IONAMENTO	103
13.1.1	3 A PARTIÇÃO DE SWAP	106
13.1.1	4 REPARTICIONAMENTO	106
13.1.1	5 FAZENDO CÓPIA DE SEGURANÇA DE SEU SISTEMA ANTIGO	107
13.1.1	6 FIPS.EXE	107
13.1.1	7 Preparando o Linux para inicializar	108
13.1.1	8 CRIANDO UM DISQUETE DE INICIALIZAÇÃO DO LINUX NO DOS	108
13.1.1	9 CRIANDO UM DISQUETE DE INICIALIZAÇÃO NO LINUX	109
13.1.2	0 PARTICIONANDO O DISCO RÍGIDO: FDISK E CFDISK	109
14 DI	STRIBUIÇÕES LINUX	112
14.1	DEBIAN GNU/LINUX	112
14.1.1	CARACTERÍSTICAS DA IN STALAÇÃO DO DEBIAN GNU/LINUX	112
14.1.2	OBTENDO AS IMAGENS DO DISQUETE	112
14.1.3	FAZENDO O DOWNLOAD DOS PACOTES	113
14.1.4	INICIALIZANDO DOS DIS QUETES E INSTALANDO O DEBIAN GNU/LINUX	114
14.1.4	.1 O disco de rescue	114
14.1.4	.2 A caixa de diálogos colorido ou monocromático	115
14.1.4	.3 O Menu Principal	115
14.1.4	.4 Configuração do teclado	116
14.1.4	.5 O shell	116
14.1.4	.6 Particionando seus discos	117
14.1.4	.7 Inicializando e Ativando a Partição de Swap	117

14.1.4.8 Inicializando uma partição Linux	117
14.1.4.9 Instalando o sistema base	118
14.1.4.10 Instalando o kernel do sistema operacional	118
14.1.4.11 Instalando os drivers de dispositivos	118
14.1.4.12 Configurando o sistema base	119
14.1.4.13 Configurando a rede	119
14.1.4.14 Tornando seu disco rígido inicializável	120
14.1.4.15 Criando um disquete de inicialização	120
14.1.4.16 O momento da verdade	120
14.1.4.17 Add a user account and password	120
14.1.4.18 Log in	121
14.1.5 RODANDO O DEBIAN GNU/LINUX	121
14.1.5.1 Classificação dos Pacotes	121
14.1.5.2 Relacionamento entre Pacotes	122
14.1.6 DSELECT	122
14.1.6.1 Access (Acesso)	123
14.1.6.2 Update (Atualização)	123
14.1.6.3 Select (Selecionar)	123
14.1.6.4 Install (Instalar)	124
14.1.6.5 Configure (Configurar)	124
14.1.6.6 Remove (Remover)	124
14.1.6.7 Quit	124
14.1.7 DPKG	124
14.1.7.1 Instalando ou atualizando pacotes novos ou existentes	124
14.1.7.2 Configurando pacotes instalados	125
14.1.7.3 Removendo pacotes instalados	125
14.1.7.4 Relatando a situação dos pacotes	125
14.1.7.5 Listando pacotes disponívies	125
14.1.7.6 Listando arquivos pertencentes por pacotes	126
14.1.7.7 Encontrando o pacote que detem um arquivo	126
14.1.7.8 Sumário	126
14.1.8 Sobre o Debian GNU/Linux	126
14.2 LINUX RED HAT	127
14.2.1 Propriedades da instalação do Linux Red Hat Linux	127
14.2.2 O SISTEMA DE GERENCIA MENTO DE PACOTES RPM	127
14.2.2.1 Convenções de nome dos pacotes	128
14.2.3 Uma nota sobre atualização do Red Hat	128
14.2.4 Criando disquetes de instalação	128
14.2.5 MÍDIA DE INSTALAÇÃO	129
14.2.5.1 Instalação por NFS	129
14.2.5.2 Instalação por disco rígido	129

14.2.6 PERSONALIZANDO SUA IN STALAÇÃO POR DISCO R ÍGIDO OU NFS	130
14.2.6.1 O arquivo comps	
14.2.7 INSTALAÇÃO MÍNIMA REC OMENDADA	132
14.2.8 QUANTO ESPAÇO VOCÊ RE ALMENTE PRECISA?	133
14.2.9 Instalação	133
14.2.10 AINDA SOBRE A MÍDIA DE INSTALAÇÃO	133
14.2.10.1 Instalação por CD-ROM	134
14.2.10.2 Instalação a partir do disco rígido	134
14.2.10.3 Instalação por NFS	134
14.2.10.4 Instalação por FTP	135
14.2.11 PASSANDO PELO RESTO DA INSTALAÇÃO	135
14.2.12 Após a instalação	137
14.2.12.1 Entendendo o prompt do LILO	137
14.2.12.2 Logando no sistema pela primeira vez	137
14.3 SLACKWARE	138
14.3.1 O SLACKWARE NÃO É PARA VOCÊ. (OU TALVEZ SEJA.)	
14.3.2 Uma breve história	138
14.3.3 POR QUÊ, ENTÃO?	138
14.3.4 Atualização? Pense duas vezes!	139
14.3.5 SELECIONE UM MÉTODO DE INSTALAÇÃO	139
14.3.6 DISCOS DE INICIALIZAÇÃO: SEMPRE UMA BOA IDÉIA	139
14.3.7 PLANILHA PARA INSTALA ÇÃO DO SLACKWARE	140
14.3.8 FAZENDO O SLACKWARE ACONTECER	146
14.3.9 Criando alguns discos de Boot	147
14.3.9.1 Escolha seu kernel!	147
14.3.10 Entrando em ação	150
14.3.11 O PROGRAMA DE INSTALAÇÃO (SETUP) SLACKWARE	150
14.3.12 Isso é tudo?	151
14.3.13 SOLUCIONANDO PROBLEMAS DE INSTALAÇÃO	151
14.3.13.1 FAQs de instalação do Slackware	151
14.3.13.2 Suporte na Web para o Slackware	151
14.3.13.3 Grupos de News Usenet Para Slackware	151
14.3.13.4 Listas de discussão sobre Slackware	152
14.3.13.5 Você recebe pelo que paga (suporte comercial)	152
14.3.14 Não Baixe a Guarda	152
14.3.15 CONSIDERE A REINSTALAÇÃO!	152
14.3.16 TORNE O SISTEMA SEGUR O	152
14.3.16.1 Saia da rede de uma vez	152
14.3.16.2 Cadastre uma senha de root	152
14.3.16.3 Crie uma conta para você	153
14.3.16.4 Proiba Logins de root	153

14.3.16.5 Verifique se há patches disponívies	153
14.3.16.6 Mantenha-se informado	153
14.3.17 BACK UP	154
15 ENFRENTANDO DIFICULDADES DE INSTALAÇÃO	155
15.1 PROBLEMAS INICIALIZAN DO ATRAVÉS DA MÍDIA DE INSTALAÇÃO	155
15.2 PROBLEMAS DE HARDWARE	157
15.2.1 ISOLANDO PROBLEMAS DE HARDWARE	157
15.2.2 PROBLEMAS NO RECONHEC IMENTO DE UM DISCO RÍGIDO OU CONTROLAD	OR159
15.2.3 PROBLEMAS COM DISPOSITIVOS E CONTROLADORE S SCSI	160
15.3 PROBLEMAS INSTALANDO SOFTWARE	161
15.4 PROBLEMAS APÓS INSTAL AR O LINUX	163
15.4.1 PROBLEMAS AO INICIALI ZAR O LINUX DO DISQUETE	163
15.4.2 PROBLEMAS INICIALIZANDO O LINUX DE SEU DISCO RÍ GIDO	163
15.4.3 PROBLEMAS PARA ENTRAR NO SISTEMA (LOGIN)	165
15.4.4 Problemas usando o sistema	165

1. INTRODUÇÃO

O Linux é possivelmente o mais importante software livre produzido desde o "Space War" ou, mais recentemente, o Emacs. Ele evoluiu para um sistema operacional completo para negócios, educação e uso pessoal. O Linux não esta mais restrito ao universo de *hackers* que ficam horas na frente de um monitor (embora muitos de seus usuários se enquadrem neste grupo).

O Linux é um sistema operacional UNIX que executa em uma variedade de plataformas, principalmente computadores pessoais com processadores 80386 ou seus sucessores. Ele suporta uma vasta gama de softwares, como o TeX, o Sistema X-Windows, o compilador GNU C/C++, e o TCP/IP. É uma implementação versátil e confiável do UNIX, distribuída livremente nos termos da Licença Geral de Uso GNU (GNU General Public License).

O Linux pode transformar qualquer computador 386 ou superior em uma estação de trabalho que coloca em suas mãos todo o poder e flexibilidade do UNIX. Empresas instalam Linux em suas redes de computadores, e usam o sistema operacional para gerenciar sistemas financeiros e hospitalares, ambientes de computação distribuída e telecomunicações. Universidades do mundo inteiro utilizam o Linux para ministrar cursos de definição e implementação de sistemas operacionais. Entusiastas por computadores usam o Linux em casa para programação, como ferramenta de produtividade, e uma infinidade de outros usos.

O que torna o Linux tão diferente de outras opções de sistema operacional é que ele é uma implementação gratuita do UNIX. Ele foi e ainda está sendo desenvolvido cooperativamente por um grupo de voluntários, principalmente através da Internet, que trocam códigos, relatam bugs e resolvem problemas em um ambiente aberto. Todos são bem vindos ao esforço de desenvolvimento do Linux. Tudo que é necessário é o interesse em mexer em uma implementação do UNIX e algum conhecimento de programação.

1.1 Sobre este livro

Este livro é um guia de instalação e iniciação ao Linux. O propósito é iniciar novos usuários consolidando tanto material quanto possível em um só livro. Ao invés de cobrir detalhes técnicos voláteis, que tendem a mudar rapidamente, damos a você uma excelente base para que possa descobrir mais por conta própria.

O Linux não é difícil de instalar e usar. Entretanto, assim como qualquer implementação do UNIX, há freqüentemente uma magia negra envolvida para fazer com que tudo funcione corretamente. Esperamos que este livro seja seu ingresso para a excursão no mundo Linux e que mostre quão incrível um sistema operacional pode ser.

Neste livro, cobrimos os seguintes tópicos:

• O que é o Linux? O projeto e filosofia deste sistema operacional exclusivo, e o que ele pode fazer por você.

- Detalhes da execução do Linux, incluindo sugestões e recomendações de configuração de hardware.
- Instruções específicas para instalar várias distribuições Linux, como Debian, Red Hat e Slackware.
- Um breve tutorial introdutório ao UNIX para usuários sem experiência anterior em UNIX. Esse tutorial deve fornecer material suficiente para que iniciantes encontrem seu caminho pelo sistema.

(....)

Este livro é para usuários que queiram iniciar no Linux. Não assumimos qualquer experiência prévia em UNIX, mas esperamos que os iniciantes busquem outras fontes de referência ao longo do aprendizado. (.....). Em geral, este livro é feito para ser lido juntamente com outro livro sobre conceitos básicos de UNIX.

1.2 Uma Breve História do Linux

O UNIX é um dos sistemas operacionais mais populares do mundo devido à enorme base instalada do mesmo. Ele foi originalmente desenvolvido na década de 70 na AT&T como um sistema multitarefa para minicomputadores e *mainframes*, mas desde então cresceu para se tornar um dos sistemas operacionais mais utilizados, apesar de sua interface às vezes confusa e a falta de uma padronização centralizada.

Muitos hackers acreditam que o UNIX é a "coisa certa", o único sistema operacional de verdade. Daí, conseqüentemente, o desenvolvimento do Linux se dá, em grande parte, por uma comunidade de hackers que desejam "sujar" suas mãos mexendo no seu próprio sistema operacional.

Versões do UNIX existem em muitos sistemas, de computadores pessoais a supercomputadores como o Cray. A maioria das versões do UNIX para computadores pessoas, porém, são caras e difíceis de utilizar.

O Linux é uma versão gratuita do UNIX desenvolvida inicialmente por Linus Torvalds na Universidade de Helsinque, na Finlândia, com a ajuda de muitos programadores UNIX espalhados pela Internet. Qualquer um com conhecimento suficiente e vontade pode alterar e melhorar o sistema. O kernel (núcleo) do Linux não utiliza nenhum código da AT&T ou de qualquer outra fonte proprietária, e muito do software disponível para Linux foi desenvolvido como parte do projeto GNU da Free Software Foundation, em Cambridge, Massachusetts, nos Estados Unidos. Entretanto, programadores do mundo inteiro contribuíram para o crescente conjunto de softwares para Linux.

O Linux foi originalmente desenvolvido como um projeto de passatempo por Linus Torvalds. Foi inspirado no Minix, um pequeno sistema UNIX desenvolvido por Andy Tanenbaum. As primeiras discussões sobre o Linux se realizaram no grupo de discussão da Usenet comp.os.minix. Estas discussões eram centradas no desenvolvimento de uma versão acadêmica do UNIX para usuários do Minix que queriam um sistema operacional com mais recursos.

O desenvolvimento inicial do Linux - na época feito todo em *assembly* - lidou principalmente com os recursos multitarefa da interface de modo protegido do 80386. Linus escreveu:

"Depois desta etapa ficou bem mais simples: um código complicado, ainda, mas a depuração era bem mais fácil. Eu comecei a utilizar C neste ponto, e isto certamente acelerou o desenvolvimento. Foi também então que comecei a pensar seriamente nas minhas idéias megalomaníacas de criar um Minix melhor que o Minix. Eu tinha a esperança de que conseguiria recompilar o gcc em cima do Linux algum dia."

"Foram pouco mais de dois meses para eu ter um driver de disco rígido (cheio de bugs, mas que pelo menos na minha máquina funcionava) e um pequeno sistema de arquivos. Foi nesta época que liberei a versão 0.01 (perto do final de agosto de 1991): não era algo bonito, não tinha driver para disquete e não podia fazer muita coisa. Duvido que alguém tivesse chegado a compilar esta versão. Mas então eu já estava envolvido, e não iria parar enquanto não superasse o Minix".

Nenhum anúncio jamais foi feito do lançamento da versão 0.01. Os códigos fonte não eram sequer executáveis. Continham apenas os rudimentos do kernel e assumiam que você tinha acesso a uma máquina Minix para compilar e experimentar com eles.

Em outubro de 1991 Linus anunciou a primeira versão "oficial" do Linux, que era a versão 0.02. Nesta fase, Linus conseguia rodar o bash (o Bourne Again Shell do GNU) e o GCC (o compilador C do GNU), mas bem pouco além disto. Novamente, tratava-se de um sistema para hackers. O foco principal era o desenvolvimento do kernel - questões como suporte a usuários, documentação e distribuição nem haviam sido levantadas. Hoje, a comunidade Linux ainda tende a tratar estas questões como secundárias perante a "verdadeira programação": o desenvolvimento do kernel.

Como Linus escreveu no comp.os.minix:

"Você ainda sente falta dos saudosos dias do Minix-1.1, quando os homens eram homens e escreviam seus próprios drivers de dispositivo? Está sem um projeto legal e morrendo de vontade de cravar seus dentes em um sistema operacional que você pode modificar como quiser? Está ficando frustrado quando tudo funciona no Minix? Com saudades de perder noites tentando fazer um programa difícil funcionar? Então esta mensagem é para você."

"Como havia mencionado um mês atrás, eu estou trabalhando numa versão gratuita de um sistema parecido com o Minix para computadores 386. Ele finalmente atingiu o estágio em que talvez seja até usável (embora talvez não, dependendo do que você quer), e eu estou disposto a disponibilizar os fontes para uma distribuição mais abrangente. É apenas a versão 0.02, mas consegui com sucesso executar nele o bash, gcc, gnu-make, gnu-sed, compress, etc."

Após a versão 0.03, Linus subiu o número de versão para 0.10, já que mais pessoas estavam trabalhando no projeto. Depois de várias outras revisões, Linus subiu o número de versão para 0.95 em março de 1992, refletindo suas expectativas que o sistema estava quase pronto para um lançamento oficial. (De modo geral, um software não recebe o número de versão 1.0 até que esteja teoricamente completo ou livre de *bugs*). Quase um ano e meio depois, no final de dezembro de 1993, o kernel do Linux ainda estava na versão 0.99.pl14, lentamente se aproximando de 1.0. Hoje, o Linux já está na versão 2.4.18, e a versão 2.5 já está em desenvolvimento.

A maioria dos principais pacotes gratuitos UNIX já foi portada para o Linux, e softwares comerciais também estão disponíveis. Mais hardware é suportado do que na versão original do kernel. Quem iria imaginar que este "pequeno" clone do UNIX iria crescer até atingir a posição que atingiu no universo de computadores pessoais.

1.3 Recursos do Sistema

O Linux suporta os recursos normalmente encontrados em outras implementações do UNIX, e muitos outros que não são encontrados em nenhum outro sistema operacional. Nesta seção iremos apresentar rapidamente os recursos do kernel do Linux.

O Linux é um sistema operacional multiusuário e multitarefa completo, como o são todas as outras versões do UNIX. Isto significa que muitos usuários diferentes podem entrar e executar programas na mesma máquina simultaneamente.

O sistema Linux é quase totalmente compatível com diversos padrões UNIX no código fonte, incluindo IEEE POSIX 1, UNIX System V e Berkeley System Distribution UNIX (BSD UNIX). O Linux foi desenvolvido tendo em mente a portabilidade do código fonte, e é comum encontrar recursos que são compartilhados por mais de uma plataforma. Boa parte do software livre UNIX disponível na Internet compila diretamente em Linux. Adicionalmente, todo o código fonte do Linux, incluindo Kernel, *drivers* de dispositivo, bibliotecas, programas para usuários e ferramentas de desenvolvimento, pode ser livremente distribuído.

Outros recursos internos específicos do Linux incluem controle de tarefas POSIX (usado em shells como csh e bash), pseudo-terminais (dispositivos tty), e suporte para dispositivos lidos dinamicamente para controle de teclados nacionais ou customizados. Linux também suporta consoles virtuais que permitem que você troque entre sessões de usuário no mesmo console de sistema. O Kernel também pode emular instruções de ponto flutuante do 387, de forma que máquinas sem um co-processador aritmético podem rodar programas que exijam este tipo de instruções.

O Linux também suporta diversos sistemas de arquivo para armazenar dados, como o ext2, que foi desenvolvido especificamente para o Linux. Os sistemas de arquivos Xenix e UNIX System V também são suportados, bem como o FAT do MS-DOS e o VFAT do Windows 95. Também são suportados os sistemas de arquivos ISO9660, para CD-ROM, e UDF, para DVD.

O Linux fornece uma implementação completa do software de TCP/IP, incluindo drivers de dispositivo para várias placas Ethernet, protocolos SLIP e PPP, para conexão TCP/IP discada, PLIP e NFS. A gama completa de clientes e serviços TCP/IP também é suportada, o que inclui FTP, telnet, NNTP e SMTP.

O kernel do Linux é desenvolvido para utilizar os recursos do modo protegido do Intel 80386 e seus sucessores. Qualquer pessoa familiarizada com o modo protegido do 80386 sabe que o chip foi desenvolvido para sistemas multitarefa, como o Linux.

O kernel suporta executáveis lidos sob demanda. Apenas aqueles segmentos do programa que eventualmente estão em uso são lidos do disco para a memória. Além disto, se diversas instâncias de um programa estão rodando ao mesmo tempo, elas compartilham a mesma memória física, o que reduz o consumo total de memória.

Visando aumentar a memória disponível, o Linux também implementa paginação de disco. Até um gigabyte de área de swap pode ser alocada no disco. Quando o sistema exige mais memória física, ele salva as páginas inativas em disco, permitindo que você rode aplicações maiores e suporte mais usuários. Entretanto, naturalmente há uma perda de velocidade, pois o disco é muito mais lento do que a memória RAM.

O kernel do Linux também implementa um gerenciamento de memória unificado para programas de usuário e *cache* de disco. Toda a memória livre é utilizada para cache de disco, que é reduzido quando são executados programas grandes.

Executáveis usam bibliotecas dinâmicas compartilhadas, de forma semelhante ao mecanismo de compartilhamento de bibliotecas do sistema operacional SunOS. Conseqüentemente, arquivos executáveis ocupam menos espaço em disco, em particular aqueles que utilizam muitas funções de biblioteca. Existem também bibliotecas lincadas estaticamente, para depuração e criação de arquivos binários completos, para o caso de sistemas que não tenham as bibliotecas dinâmicas instaladas.

Para facilitar a depuração, o kernel gera *core dumps* (imagens da memória) para análise em caso de falha do sistema, tornando possível identificar o que fez determinado programa derrubar o sistema.

1.4 Recursos de Software

Praticamente todas as ferramentas que se espera encontrar em uma implementação padrão do UNIX já foram portadas para o Linux, incluindo os comandos básicos, como ls, awk, tr, sed, bc, e more. O ambiente de desenvolvimento de outros sistemas UNIX é duplicado no Linux. Todos os comandos e utilitários padrões estão incluídos.

Muitos editores de texto estão disponíveis, como vi, ex, pico, jove e GNU Emacs. O editor de texto que você está acostumado a usar muito provavelmente já foi portado para o Linux.

A escolha do editor de texto é uma questão interessante. Muitos

usuários UNIX preferem editores "simples" como vi, mas vi tem muitas limitações devido a sua idade, e editores modernos como o Emacs tem ganhado popularidade. O Emacs suporta uma linguagem macro e um interpretador baseados em Lisp, poderosas sintaxes de comando e outras extensões. Existem pacotes de macros Emacs que permitem que você leia emails, edite o conteúdo de diretórios, etc.

A maioria dos aplicativos básicos do Linux são softwares GNU. Aplicativos GNU suportam recursos avançados não encontrados nas versões padrão de programas BSD ou UNIX System V. Por exemplo, o clone GNU do vi, o elvis, inclui uma linguagem de macro estrutura que é diferente da versão original. Entretanto, os aplicativos GNU permanecem compatíveis com as versões BSD e System V. Muitas pessoas consideram as versões GNU superiores às originais.

Um shell é um programa que lê e executa comandos do usuário. Além disto, muitos programas shell oferecem recursos como controle de tarefas, gerenciamento simultâneo de diversos processos, redirecionamento de entrada e saída e uma linguagem de comando para escrita de scripts shell. Um script shell é um programa na linguagem de programação do shell e é análogo aos arquivos batch do MS-DOS.

Muitos tipos de shell estão disponíveis no Linux. A mais importante diferença entre shells é a linguagem de programação. Por exemplo, o C Shell (csh) usa uma linguagem de comando similar a linguagem de programação C. O Bourne Shell (sh) usa outra linguagem de comando. A escolha do shell é geralmente baseada na linguagem de comandos que ele oferece e determina significativamente a qualidade do ambiente de desenvolvimento que estará sendo oferecido naquele Linux.

O GNU Bourne Again Shell (bash) é uma variação do Bourne Shell que inclui muitos recursos avançados como controle de tarefas, histórico de comandos, uma interface semelhante ao Emacs para edição de linhas de comando e outras poderosas extensões do Bourne Shell padrão. Outro shell popular é o tcsh, uma versão do C Shell com avanços de funcionalidade semelhantes ao bash. Outros shells incluem zsh, ksn, ash e rc.

(.....)

1.4.1 Ferramentas e Linguagens de Programação

O Linux fornece um ambiente completo de programação UNIX, incluindo todas as bibliotecas padrão, ferramentas de programação, compiladores e ferramentas de depuração.

São suportados padrões como o POSIX.1, o que permite que programas escritos para o Linux sejam facilmente portados para outros sistemas operacionais. Programadores profissionais UNIX e administradores de sistema usam o Linux para desenvolver programas em casa, então copiam o programa para outras plataformas UNIX. Estudantes de computação aprendem programação UNIX e exploram outros aspectos do sistema, como a arquitetura do kernel, com Linux. Com o Linux, o programador tem acesso ao fonte do kernel e de todas as bibliotecas.

Dentro do mundo de software UNIX, sistemas e aplicações são geralmente programados em C ou C++. O compilador C e C++ padrão no Linux é o GNU gcc, que é um compilador avançado que suporta C++, incluindo recursos do ANSI/ISO C++, bem como Objective-C, outra linguagem orientada a objetos derivada do C.

Além de C e C++, outras linguagens compiladas e interpretadas já foram portadas para o Linux, como Smalltalk, FORTRAN, Java, Pascal, Lisp, Scheme e ADA. Adicionalmente, várias ferramentas para programação em Assembly, estão disponíveis, bem como interpretadores Perl e Tcl/Tk.

O depurador gdb pode avançar pelo código fonte de um programa, uma linha por vez, e examinar um *core dump* para encontrar a causa de uma queda do programa. A ferramenta gprof permite análises estatísticas de desempenho, informando as partes em que o programa gasta mais tempo de execução. Como mencionado anteriormente, o editor de texto Emacs fornece um ambiente de programação que permite a edição e compilação de fontes em várias linguagens.

Outras ferramentas incluem o GNU make e GNU image, que permitem manipular a compilação de grandes aplicações, e o RCS, um sistema para controle de versões de código fonte.

Finalmente, o Linux suporta DLLs, que são bibliotecas compartilhadas e dinamicamente lincadas, e permitem executáveis muito menores.

1.4.2 Introdução ao Sistema X-Windows

O Sistema X-Windows, ou simplesmente X, é uma poderosa interface gráfica (GUI - Graphical User Interface) para máquinas UNIX e é um poderoso ambiente que suporta muitas aplicações. Usando o Sistema X-Windows, você pode ter diversas janelas de terminal na tela ao mesmo tempo, cada uma tendo uma sessão de usuário diferente. Um mouse geralmente é utilizado com o X, ainda que não seja indispensável.

Muitas aplicações específicas do ambiente X já foram escritas, incluindo jogos, ferramentas gráficas e de programação e ferramentas de documentação. Através de uma rede TCP/IP, uma máquina Linux pode mostrar aplicações X que estejam rodando em outras máquinas.

O Sistema X-Windows foi originalmente desenvolvido no MIT (Massachusetts Institute of Technology) e é gratuito. Muitas empresas já distribuíram versões proprietárias aprimoradas do Sistema X-Windows. A versão do X para Linux é o XFree86, uma versão portada do X11R6. O XFree86 suporta uma larga gama de adaptadores de vídeo, incluindo VGA, Super VGA e placas aceleradoras de vídeo. O XFree86 inclui o servidor X, aplicações, utilitários, bibliotecas de programação e documentação.

Aplicações padrão do X incluem o xterm, um emulador de terminal utilizado para executar aplicações de modo texto em uma janela, entre outros. Entre as muitas aplicações em X estão incluídas planilhas eletrônicas, editores de texto, ferramentas gráficas e navegadores da Web, como o Netscape. Em teoria, qualquer aplicação desenvolvida para o X irá compilar e executar no Linux.

A interface do Sistema X-Windows é controlada pelo gerenciador de

janelas. Este programa é responsável pelo posicionamento, movimentação e redimensionamento das janelas, controle dos ícones associados às janelas, entre outras tarefas semelhantes. O XFree86 inclui o twm, o gerenciador de janelas clássico do MIT, e gerenciadores de janela avançados, como o Open Look Virtual Window Manager (olvwm). Outro gerenciador de janelas popular é o fvwm, que mostra as janelas com uma aparência tridimensional. Muitas distribuições do Linux utilizam o fvwm como gerenciador padrão de janelas. Uma versão do fvwm chamada fvwm95-2 imita o visual do ambiente de janelas Windows.

A distribuição do XFree86 inclui bibliotecas de programação para programadores que desejam desenvolver aplicações para o X.

A exigência mínima para utilizar o X é um 80386 com pelo menos 4 megabytes de RAM, embora 16 MB ou mais seja recomendável.

1.4.3 Introdução às Redes

Você gostaria de se comunicar com o mundo? Linux suporta dois protocolos de rede UNIX: TCP/IP e UUCP. TCP/IP (Transmission Control Protocol/Internet Protocol) é o paradigma de conexão que permite que sistemas de todo o mundo se comuniquem em uma rede única, a Internet. Com Linux, TCP/IP e uma conexão com a Internet, você pode se comunicar com usuários e máquinas através de e-mail, news e FTP.

A maioria das redes TCP/IP utiliza a Ethernet como meio físico de transporte. O Linux suporta várias placas Ethernet para computadores pessoais, incluindo adaptadores PCMCIA.

Entretanto, como nem todos tem uma conexão Ethernet em casa, Linux também suporta os protocolos SLIP e PPP, que permitem o acesso a Internet via modem. Muitas empresas e universidades possuem servidores SLIP e PPP. Se seu sistema Linux tem uma conexão Ethernet com a Internet, você pode torná-lo também um servidor SLIP ou PPP.

O NFS (Network File System) permite que seu sistema Linux compartilhe arquivos e diretórios com outras máquinas da rede. O FTP permite que você transfira arquivos de e para outras máquinas. O Sendmail envia e recebe emails através do protocolo SMTP. Telnet, rlogin e rsh permitem que você entre em outras máquinas presentes em uma rede e execute comandos nelas. O finger permite que você recupere informações sobre outros usuários da Internet.

O Linux também suporta conexão com máquinas Windows através do Samba e com máquinas Macintosh com AppleTalk e LocalTalk. Suporte ao protocolo IPX da Novell também está incluído.

A gama completa de leitores de e-mail e news também está disponível para o Linux, incluindo elm, pine, rn, nn e tin. Qualquer que seja sua preferência, você pode configurar um sistema Linux para enviar e receber news e e-mail para qualquer lugar do mundo.

O Linux fornece ainda uma interface padrão para a programação de sockets. Virtualmente qualquer programa que utiliza TCP/IP pode ser portado para o Linux. O servidor X do Linux também suporta TCP/IP, e aplicações rodando remotamente podem utilizar o monitor de seu computador.

UUCP (UNIT-to-UNIX Copy) é um mecanismo mais antigo de transferência de arquivos, e-mails e news entre máquinas UNIX. Historicamente, máquinas UUCP se conectam por linhas telefônicas, via modem, mas UUCP pode ser utilizado também para transferir dados em uma rede.

1.4.4 Telecomunicações e Softwares de BBS

Se você tem um modem, você será capaz de se comunicar com outras máquinas via pacotes de telecomunicação disponíveis para o Linux. Um pacote popular de comunicação para o Linux é o seyon, que fornece uma interface customizável e ergonômica no X, e possui suporte para os protocolos de transferência de arquivos Kermit e ZModem. Outros programas de telecomunicação incluem o C-Kermit, pcomm e minicom. São programas similares aos encontrados em outros sistemas operacionais e relativamente fáceis de utilizar.

Se você não tem acesso a um servidor SLIP ou PPP, pode utilizar o term para dividir sua linha serial. O programa term permite que você abra mais de uma sessão de usuário em uma conexão por modem. Permite ainda que você redirecione conexões de um cliente X para seu servidor X local por uma linha serial. Outro pacote de software, KA9Q, implementa uma interface SLIP similar.

1.4.5 World Wide Web

É importante observar que o Linux inclui servidores web, bem como navegadores da web. O servidor mais conhecido é o Apache. Milhares de sistemas Linux rodam Apache na Internet hoje em dia.

Distribuições do Linux incluem diferentes web browsers, e outros podem ser copiados da Internet. Browsers disponíveis incluem Lynx, Mosaic, Netscape, Arena e Amaya.

O Linux também oferece suporte completo para Java e CGI, e o Perl é uma ferramenta padrão do ambiente de programação Linux.

1.4.6 Interface com o MS-DOS

Existem diversas ferramentas que fazem interface com o MS-DOS. A mais conhecida é o Linux MS-DOS Emulator, que permite que se execute aplicativos MS-DOS diretamente do Linux. Ainda que ambos sejam sistemas operacionais completamente diferentes, o ambiente de modo protegido do 80386 possibilita que aplicações MS-DOS comportem-se como se estivessem executando diretamente.

Diversas ferramentas executam no emulador MS-DOS. Naturalmente, aplicações MS-DOS que usam recursos bizarros e esotéricos do sistema podem nunca vir a ser suportadas, devido às limitações inerentes a qualquer emulador.

Comandos e ferramentas que são padrão no MS-DOS, como pkzip.exe, 4dos, e outras, executam normalmente no emulador.

O emulador MS-DOS se destina a ser apenas uma solução para quem precisa utilizar algumas poucas aplicações MS-DOS, e utiliza Linux para todas as outras atividades. Não se destina a ser uma implementação completa do MS-DOS. Se o emulador não é suficiente para atender as suas necessidades, sempre é possível executar o MS-DOS e o Linux na mesma máquina. Utilizando o LILO, você pode especificar na inicialização do sistema qual o sistema operacional que vai ser executado.

Outro emulador, o WINE, se destina a permitir o uso de aplicativos MS-Windows, emulando o Microsoft Windows no Sistema X-Windows.

1.4.7 Outras Ferramentas

Um conjunto de outros programas e ferramentas existe para Linux. Bancos de dados relacionais, como Postgres, Ingres e Mbase, bem como bancos de dados comerciais, como Oracle, existem em plataforma Linux.

Muitas outras aplicações já foram portadas para Linux. Se você não consegue encontrar o que procura, você pode você mesmo portar uma aplicação de outra plataforma para Linux.

(.....)

1.5 Questões de Copyright

O Linux é protegido pelo que é conhecido como a GNU GPL (GNU General Public License). A GPL foi desenvolvida pela Free Software Foundation para o projeto GNU e especifica uma série de regras para a distribuição e modificação de software livre. Livre, neste sentido, refere-se a distribuição, não ao custo.

Originalmente, Linus Torvalds disponibilizou o Linux através de uma licença mais restritiva que a GPL, que permitia que o software fosse livremente distribuído e modificado, mas que impedia que qualquer quantia em dinheiro fosse cobrada por sua distribuição e uso. A GPL, por outro lado, permite que pessoas vendam e tenham lucro com software livre, mas não permite que elas restrinjam de qualquer forma o direito de outros de distribuírem também o software como bem entenderem.

Primeiro é preciso que se explique que o software livre protegido pela GPL não é de domínio público. Software de domínio público, por definição, não tem um copyright e pertence ao público. Software protegido pela GPL, por outro lado, é copyright do autor. O software é protegido pelas leis internacionais de copyright, e o autor está legalmente definido. A GPL protege softwares que podem ser livremente distribuídos, mas que não são de domínio público.

Software GPL também não é shareware. Normalmente, sharewares pertencem a um autor que exige que os usuários lhe enviem dinheiro para utilizá-lo. Softwares GPL podem ser distribuídos e utilizados sem que nada seja cobrado por eles.

A GPL também permite que qualquer um utilize, modifique e distribua suas próprias versões do software. Entretanto, todo e qualquer trabalho derivado de um software protegido pela GPL precisa, necessariamente, também ele ser protegido pela mesma licença. Em outras palavras, uma empresa não pode modificar o Linux e vendê-lo protegido por uma licença mais restritiva que a GPL. Se o software deriva do Linux, ele tem que ser associado a GPL. Entretanto, você pode revender qualquer produto derivado do Linux, desde que o mesmo também esteja disponível gratuitamente.

A primeira vista, isto pode parecer uma contradição. Porque vender software quando a GPL obriga que o mesmo esteja também disponível gratuitamente? Imagine que uma empresa grave um CD-ROM com diversos softwares gratuitos. Obviamente ela teria um custo por este trabalho que teria que ser cobrado dos interessados em adquirir o produto. A GPL permite que ela cubra os custos e mesmo tenha lucro vendendo um CD-ROM com software livre.

Organizações que vendem software livre têm, porém, que seguir certas restrições definidas na GPL, não podendo restringir os direitos dos usuários que adquirem o produto. Se você compra um CD-ROM que contenha produtos protegidos pela GPL, você pode copiar e distribuir o CD-ROM sem ter que pagar nada por isto, e mesmo revendê-lo você mesmo. Os distribuidores têm que deixar claro para seus compradores que os softwares estão protegidos pela GPL e disponibilizar o código-fonte dos mesmos sem nenhum custo. Isto permite que qualquer um que compre software protegido pela GPL possa modificá-lo.

Permitir que empresas distribuam ou vendam software livre é algo importante, pois nem todos têm acesso a Internet e condições de baixar software de graça. Muitas organizações vendem Linux em CD-ROM e outras formas, e lucram com as vendas. Desenvolvedores Linux provavelmente nunca vão receber uma parcela deste lucro, mas este é um entendimento obtido entre os programadores e distribuidores de softwares que obedecem a GPL. Em outras palavras, Linus Torvalds sabia que empresas poderiam vender Linux e que ele poderia não ganhar um centavo com estas vendas.

No mundo do software livre, a questão fundamental não é o dinheiro. O objetivo do software livre é sempre de desenvolver e distribuir programas excelentes e permitir que qualquer um obtenha e use. Na próxima seção, nós iremos discutir como isto se aplica ao desenvolvimento do Linux.

1.6 O Projeto e a Filosofia do Linux

Novos usuários geralmente têm algumas falsas expectativas sobre o Linux. É importante entender a filosofia e o projeto do Linux para usá-lo efetivamente. Nós iremos começar apresentando qual *não* é a filosofia dele.

Em empresas que desenvolvem versões comerciais do UNIX, o sistema todo é desenvolvido segundo uma política rigorosa de controle de qualidade, que utiliza sistemas de controle de códigos fonte, documentação e procedimentos para relatar e resolver bugs. Desenvolvedores não podem alterar recursos e mudar seções chaves do código ao seu bel prazer. Eles precisam justificar a mudança como resposta a um relato de *bug* e em seguida marcar todas as mudanças no sistema de controle do código fonte, de tal forma que as mudanças possam ser desfeitas se necessário. Cada desenvolvedor é responsável por uma ou mais partes do código, e apenas o responsável pode

fazer alterações em sua seção do código fonte.

Um departamento de controle de qualidade executa rigorosos testes em cada nova versão do sistema operacional e relata qualquer bug. Os desenvolvedores consertam os bugs relatados. Um complexo sistema de análise estatística é utilizado para garantir que uma certa percentagem dos bugs seja resolvida antes da liberação da próxima versão do sistema operacional.

A empresa, naturalmente, precisa ter alguma prova quantificável que a próxima versão do sistema operacional está pronta para ser liberada, o que torna necessária a coleta de análises estatísticas quanto ao desempenho do mesmo. É um trabalho de grandes proporções desenvolver um sistema UNIX comercial, geralmente empregando centenas, ou mesmo milhares, de profissionais. Obviamente, não existem dois fabricantes de versões comerciais do UNIX que sejam exatamente iguais, mas este é o cenário típico.

O modelo de desenvolvimento do Linux descarta todo o conceito clássico de desenvolvimento organizado, sistemas de controle de código fonte, relatórios estruturados de bugs e controle estatístico de qualidade. Linux é, e possivelmente sempre será, um sistema operacional desenvolvido por *hackers*. (Por *hacker*, queremos dizer um programador expert, dedicado a mexer com computadores. Este é o conceito original da palavra hacker, em contraste com a conotação de hacker como alguém que utiliza o computador com objetivos ilegais. Atualmente as palavras *cracker* ou *haxor* têm ou sido utilizadas para denominar este segundo grupo de indivíduos).

Não existe uma organização exclusivamente responsável pelo desenvolvimento do Linux. Qualquer um com conhecimento suficiente pode auxiliar no desenvolvimento e teste do kernel, no porte de novos softwares, na escrita de documentação, e mesmo no auxílio a novos usuários. Através de listas de e-mail e grupos Usenet a comunidade Linux mantém contato entre si. Qualquer um que queira incluir código na versão oficial do kernel pode enviar um e-mail para Linus Torvalds, que irá testar e decidir se é possível incluir o código no kernel.

O sistema em si é projetado de forma aberta e orientado a recursos. Os critérios para lançamento de uma nova versão incluem o número de bugs consertados, retorno para usuário durante testes pré-lançamento, etc.

É suficiente dizer que nem todos os bugs são corrigidos, nem todos os problemas são solucionados entre versões. Enquanto as revisões estiverem livres de erros críticos e bugs recorrentes, elas são consideradas estáveis.

Qualquer um que desenvolver um novo recurso ou ferramenta geralmente disponibiliza o mesmo como uma versão alfa, ou seja, uma versão de teste, para aqueles que quiserem auxiliar na descoberta de problemas no código. Versões Alfa são geralmente disponibilizadas inicialmente em determinados *sites* de ftp, e mensagens são enviadas para uma ou mais listas de Linux. Usuários que instalam as versões alfa das ferramentas enviam suas opiniões, correções de bugs e perguntas para os autores.

Depois que os primeiros bugs foram resolvidos, o código entra a fase de Beta teste, na qual é considerado estável, mas incompleto. Ele funciona, mas nem todos os recursos necessariamente estão presentes. Eventualmente, o software pode ir direto para a etapa final, na qual é considerado completo e utilizável.

Tenha em mente que estas são convenções, não regras. Alguns programadores podem se sentir tão confiantes em seus softwares que optem por não lançar versões alfa ou beta. Sempre cabe ao desenvolvedor tomar este tipo de decisão.

Você pode chegar a duvidar que tal sistema desestruturado de voluntários que programam e testam um UNIX completo consigam produzir resultados. Hoje é sabido que o Linux é um dos mais eficientes e motivados esforços de desenvolvimento já realizado. O kernel inteiro do Linux foi construído do zero, sem nenhum código proprietário. Bibliotecas são escritas e portadas, sistemas de arquivo são desenvolvidos, e drivers de dispositivos são escritos, tudo graças ao trabalho de voluntários.

O software Linux geralmente é disponibilizado como uma **distribuição**, um conjunto de ferramentas já empacotado que compreende todo um sistema. Seria difícil para a maioria dos usuários instalar um sistema completo do zero, começando com o kernel, incluindo ferramentas e instalando todos os softwares necessários, sem utilizar uma distribuição. Assim, muitas distribuições estão disponíveis que incluem tudo que é necessário para instalar e rodar um sistema completo. Não existe uma única distribuição padrão. Existem muitas, e cada uma tem suas vantagens e desvantagens.

1.7 Diferenças entre o Linux e outros Sistemas Operacionais

É importante entender as diferenças entre o Linux e outros sistemas operacionais, como MS-DOS, Windows e outras implementações de UNIX para computadores pessoais. Em primeiro lugar, Linux coexiste pacificamente com outros sistemas operacionais na mesma máquina. Você pode executar MS-DOS ou Windows na mesma máquina em que roda Linux.

1.7.1 Porque usar o Linux?

Porque utilizar Linux ao invés de um sistema operacional comercial? Para usuários e programadores de UNIX e estudantes que têm interesse em dominar tal plataforma, faz muito mais sentido utilizar um sistema operacional Linux do que Windows em casa. Para hackers, no sentido sempre de experts em informática, Linux é o sistema operacional ideal.

Para empresas, Linux pode ser uma solução mais barata, robusta e eficiente que outras alternativas. Muitas empresas têm utilizado como forma de evitar os altos custos anuais com compras de licenças de software, enquanto outras têm escolhido como forma de utilizar eficientemente máquinas ultrapassadas, que não rodam nas versões recentes do Windows, mas funcionam com eficiência em um Linux.

1.7.2 Linux vs Windows

Tanto as diferentes versões do Windows quanto o Linux são sistemas operacionais multitarefa, suportando aproximadamente os mesmos recursos de

interface com usuários, conexão com redes de computadores, etc. A verdadeira diferença é que o Linux é uma versão do UNIX, beneficiando-se da contribuição da comunidade UNIX como um todo.

Ao contrário do Linux, o Windows é propriedade de uma única empresa. O desenvolvimento do sistema operacional é controlado por uma única corporação, o que traz vantagens em termos de padronização e definição de interface, mas também significa que as definições de política de preços e padrões a serem utilizados serão todos definidos por uma empresa.

(.....)

1.7.3 Linux vs outras implementações de UNIX

Existem diversas outras implementações do UNIX para o PC, mesmo porque a arquitetura do 80386 e de seus sucessores é bastante adequada à estrutura do UNIX.

Outras implementações do UNIX para computadores pessoas são similares ao Linux. Praticamente todas as versões do UNIX suportam os mesmos softwares, ambientes de programação e recursos de conexão. Diferenças existem, porém.

Muitos usuários relatam que o Linux é pelo menos tão estável quanto versões UNIX comerciais. O tempo para o desenvolvimento de novos recursos, porém, costuma ser significativamente menor.

O mais importante fator, porém, é preço. Sendo um software livre, o Linux pode ser copiado de graça da Internet ou adquirido por um pequeno valor na forma de uma distribuição em CD-ROM. Se você pretende instalar o Linux em um grupo de máquinas, o custo é apenas o de comprar uma única cópia.

Implementações comerciais do UNIX, por outro lado, oferecem mais do que apenas o software. Em geral o preço pago pelas mesmas inclui documentação, suporte e garantias quanto a qualidade. Para grandes instituições, estas podem ser questões importantes, mas usuários domésticos e mesmo determinadas empresas podem preferir não seguir por esta alternativa e fazer uso do Linux gastando um valor muito menor.

(.....)

1.8 Exigências de Hardware

É possível que você esteja convencido das potencialidades do Linux e ansioso por aproveitar tudo o que ele tem a fazer. Entretanto, antes de sair instalando o sistema operacional, você precisa estar ciente das necessidades e limitações do mesmo em termos de hardware.

1.8.1 Placa-mãe e CPU

O Linux atualmente suporte sistemas com processadores 80386, 80486, Pentium e suas variações e sucessores, bem como outros chips compatíveis com Intel. Linux também já foi portado para PowerPC, Alpha, MIPS e Sparc.

1.8.2 Exigências de Memória

O Linux exige muito pouca memória, em comparação com outros sistemas operacionais avançados. Você precisa ter pelo menos 4MB de RAM, mas 16MB é recomendável. Quanto mais memória, mais rápido o sistema irá rodar. Algumas distribuições podem exigir mais memória RAM.

O Linux suporta o endereçamento de 32 bits do processador, utilizando toda a memória RAM automaticamente.

O Linux irá funcionar em uma máquina com 4MB de RAM, podendo inclusive executar o sistema X-Windows e o editor Emacs. Entretanto, ter mais memória é tão importante quanto ter um processador mais rápido. 16MB é suficiente para muitas aplicações, mas sistemas que vão operar como servidores podem precisar de consideravelmente mais memória, de acordo com o tipo de serviço que eles vão executar.

A maioria dos usuários Linux aloca uma porção do disco como espaço de swap, que é utilizado como uma memória RAM virtual. Mesmo que seu computador tenha 16MB ou mais, pode ser interessante utilizar espaço de swap.

1.8.3 Controladores de Disco Rígido

É teoricamente possível executar o Linux de um disquete ou, em algumas distribuições, diretamente de um CD-ROM, mas para se ter um bom desempenho é necessária uma área no disco rígido. O Linux pode co-existir com outros sistemas operacionais - ele apenas precisa de uma ou mais partições.

O Linux suporta controladores IDE e EIDE. Em geral, se o MS-DOS ou outro sistema operacional consegue acessar o disco rígido, o Linux também consegue.

1.8.4 Espaço em Disco

Obviamente, para instalar o Linux você precisa de algum espaço disponível no disco. O Linux suporta mais de um disco rígido, de tal forma que você pode instalar partes do sistema em cada disco.

O espaço em disco necessário para o Linux depende de suas necessidades e do software que você irá instalar. Para uma implementação do UNIX, o Linux é relativamente pequeno. É possível executar o sistema com apenas 20MB de espaço em disco. Naturalmente, para expansões e grandes pacotes de software, como as interfaces gráficas, mais espaço é necessário. Exigências realistas de espaço em disco variam de 200MB até 1GB ou mais.

Cada distribuição do Linux vem com uma documentação que irá ajudá-lo a definir as necessidades exatas de espaço em disco.

1.8.5 Outros Dispositivos

De uma forma geral, todos os dispositivos comumente encontrados em computadores e mesmo alguns relativamente incomuns são suportados pelo

Linux. Sempre que surge algum dispositivo novo no mercado que não tem suporte em Linux, alguém da comunidade de desenvolvedores imediatamente começa a trabalhar em um driver para o mesmo.

(.....)

1.9 Fontes de informação sobre o Linux

Existem muitas fontes de informação sobre o Linux. Em particular, livros sobre o UNIX em geral serão muito úteis, especialmente para quem não estiver familiarizado com o UNIX.

Também há muita informação online sobre o Linux. Um bom lugar para começar é o site http://www.linuxresources.com/.

1.9.1 Documentação Online

Muita documentação do Linux está disponível por FTP anônimo de sites na Internet ao redor do mundo e em redes como a Fidonet e CompuServe. Distribuições do Linux em CD-ROM também contêm documentos mencionados aqui. Se você pode mandar e-mail para a Internet, também pode pegar esses arquivos usando um dos servidors de e-mail FTP, que mandam os documentos ou arquivos dos sites de FTP por e-mail.

(.....)

1.9.2 Linux na Web

A página do Projeto de Documentação Linux (Linux Documentation Project) está em http://www.linuxdoc.org/. Esta página lista muitos "HOWTOs" e outros documentos em formato HTML, assim como links para outros sites de interesse de usuários Linux, como o ssc.com, onde está o *Linux Journal*, uma revista mensal. Sua home page está em http://www.ssc.com/.

1.9.3 Livros e outros trabalhos publicados

Os livros do Projeto de Documentação Linux são o resultado de um esforço ao redor da rede para escrever e distribuir um bom conjunto de manuais para Linux, análogos à documentação que vem com as distribuições UNIX comerciais, cobrindo instalação, operação, programação, rede, e desenvolvimento do kernel.

Os manuais do Projeto de Documentação Linux estão disponíveis por FTP anônimo e pedidos por correio.

Grandes editoras, incluindo MIS:Press, Digital Press, O'Reilly & Associates, e SAMS publicaram livros sobre o assunto. Verifique numa livraria ou na página do SSC (http://www.ssc.com/), ou ainda nas análises no Linux Journal, disponíveis no site http://www.linuxjournal.com.

Um grande número de livros sobre UNIX em geral é aplicável ao Linux. No uso e na interface de programação o Linux não difere muito de outras implementações do UNIX. Quase tudo que você gostaria de saber sobre usar e programar o Linux pode ser encontrado em textos genéricos sobre UNIX. De fato, este livro objetiva suplementar a biblioteca de livros UNIX atualmente disponível. Aqui, apresentamos os detalhes mais importantes e específicos do Linux e esperamos que você procure outras fontes de informação para aprofundar-se.

Equipado com este e outros bons livros sobre UNIX, você deve estar apto a lidar com quase tudo.

A revista *Linux Journal* é distribuída no mundo todo e é uma excelente maneira de manter-se em contato com as novidades da comunidade Linux, especialmente se você não tem acesso aos grupos de discussão da Usenet.

1.9.4 Grupos de discussão da Usenet

Usenet é um fórum de discussão eletrônico com uma seleção diversa de newsgroups (grupos de discussão), que por sua vez são áreas de discussão voltadas para um tema específico. Muitos debates sobre desenvolvimento do Linux ocorrem na Internet e Usenet. Há, portanto, um grande número de grupos de discussão dedicados ao Linux.

O primeiro grupo de discussão do Linux, alt.os.linux, foi criado para centralizar os debates sobre Linux que ocorriam em comp.os.minix e várias outras listas de e-mail. Rapidamente, o tráfego no alt.os.linux cresceu ao ponto de ser solicitada a criação de um novo grupo de discussão na hierarquia comp. Uma votação em Fevereiro de 1992 aprovou a criação do comp.os.linux.

comp.os.linux tornou-se rapidamente um dos mais populares (e barulhentos) grupos do Usenet. Mais popular que qualquer outro grupo no hierárquia comp.os.

Se você não tem acesso à Usenet, há alguns gateways de mail-to-news disponíveis para muitos dos grupos de discussão.

(.....)

1.9.5 Listas de e-mail por Internet

Se você tem acesso a e-mail, você participar de muitas listas, mesmo não tendo acesso ao Usenet. Se você não está conectado diretamente na Internet, pode se inscrever uma destas listas se puder trocar e-mail com a Internet (com UUCP, por exemplo).

Para mais informações sobre listas de e-mail sobre Linux, mande um e-mail para:

majordomo@vger.rutgers.edu

Inclua uma linha com a palavra help no corpo da mensagem e uma mensagem será retornada para você descrevendo com se cadastrar e descadastrar de várias listas de e-mail. A palavra lists em uma linha retorna o nome de várias listas que são acessíveis pelo servidor majordomo.vger.rutgers.edu

Há muitas listas de Linux com propósitos específicos. A melhor maneira de encontrar informação sobre isso é observar os newsgroups de Linux na Usenet, assim como verificar o rol de listas públicas de e-mail disponíveis, que é publicado no grupo news.answers da Usenet.

1.10 Buscando ajuda sobre Linux

Sem dúvida alguma você vai precisar de ajuda durante suas "aventuras" no mundo Linux. Até mesmo experts em UNIX ficam ocasionalmente confusos com alguma particularidade do Linux. É importante saber como, onde e quando buscar ajuda.

O meio básico de obter ajuda é através de listas de e-mail e grupos de discussão, como discutido nas seções anteriores. Se você não tem acesso a essas fontes, você pode procurar outros meios online, como BBSs e CompuServe. Também está disponível online o *Linux Journal*'s Best of Technical Support, no endereço http://www.linuxjournal.com/techsup.html.

Muitas empresas oferecem suporte comercial para o Linux. Muitos desses serviços oferecem contratos de suporte que dão acesso a consultas sobre problemas de Linux.

Mantenha em mente as sugestões abaixo para melhorar sua experiência em Linux e garantir um maior sucesso na busca de ajuda.

Consulte toda a documentação disponível primeiro! Esta é a primeira coisa a fazer quando você encontrar um problema. As documentações disponíveis foram arduamente escritas para pessoas que precisem de ajuda no sistema operacional Linux (exemplo: HOWTOs). Como mencionado anteriormente, livros escritos para UNIX são aplicáveis ao Linux e devem ser usados também.

Se você tem acesso ao Usenet, ou qualquer lista de e-mail sobre Linux, leia primeiramente a infornação que já foi publicada. Freqüentemente, soluções para problemas comuns que não estão cobertas na documentação são amplamente discutidas nas listas. Se você publicar uma pergunta sem ler antes se este assunto já foi tratado, prepare-se para as respostas:).

Aprenda a apreciar a autoconfiança. Você pediu por isso ao executar o Linux! Lembre-se, hacking tem tudo a ver com Linux. Ele não é um sistema comercial, nem pretende tornar-se um. Será esclarecedor investigar e resolver problemas por conta própria - você até poderá ser um guru Linux algum dia! Aprenda a apreciar o valor de vasculhar o sistema e consertar problemas por conta própria. Não espere, no entanto, criar um sistema Linux totalmente personalizado sem muita mão de obra.

Mantenha-se calmo. Você não vai ganhar nada usando uma marreta no seu computador Linux. Compre um bom saco de pancadas ou saia para uma longa caminhada. Essa é uma boa maneira de aliviar situações de estresse. Com a maturidade do Linux e das distribuições, esperamos que estes problemas diminuam. Entretanto, mesmo UNIX comerciais podem ser problemáticos. Quando tudo falhar, sente-se, inspire fundo e retorne ao problema quando estiver relaxado. Sua mente e consciência estarão mais

claras.

Evite publicar perguntas desnecessárias. Muitas pessoas cometem o erro de mandar mensagens pedindo ajuda prematuramente. Quando encontrar algum problema, não corra para mandar uma mensagem na lista. Primeiro, tente resolver por sua conta, e esteja absolutamente certo do problema que tem em mãos. O seu sistema não liga? Talvez esteja desligado da tomada...

Quando pedir ajuda, faça valer a pena. Lembre-se que as pessoas que vão ler sua mensagem não estão lá necessariamente para ajudar você. Lembre-se de ser educado, resumido e informativo tanto quanto possível.

Como fazer tudo isso? Primeiramente, você deve incluir toda a informação relevante possível sobre seu sistema e problema. Mandando uma mensagem simples, como: "Não consigo fazer o e-mail funcionar" provavelmente não será de grande ajuda e vai levá-lo a lugar nenhum a não ser que você inclua informação sobre seu sistema, que software usa, o que você já tentou fazer e quais os resultados que obteve. Quando incluir informações técnicas, é uma boa idéia incluir também informação sobre as versões de software utilizadas (versão do kernel, por exemplo), assim como um breve sumário da configuração de hardware. Também não exagere - o tipo de monitor e marca é provavelmente irrelevante se você está tentando configurar sua placa de rede.

2 TUTORIAL SOBRE LINUX

2.1 Introdução

Se você for novo no UNIX e Linux, pode estar um pouco intimidado pelo tamanho e aparente complexidade do sistema na sua frente. Este capítulo não entra em grandes detalhes nem trata de tópicos avançados. Ao invés disso, queremos que você comece colocando a mão na massa.

Assumimos muito pouco sobre o que você já sabe, exceto talvez que você tenha alguma familiarização com computadores, e com MS-DOS. Entretanto, mesmo que você nunca tenha usado o MS-DOS, deve ser capaz de entender tudo exposto aqui. À primeira vista, o Linux se parece bastante com o MS-DOS -- mesmo porque, partes do MS-DOS foram modeladas no sistema operacional CP/M, que por sua vez foi modelado no UNIX. No entanto, só as características mais superficiais do Linux lembram o MS-DOS. Mesmo que você seja completamente novo ao mundo do PC, este tutorial deve ajudá-lo.

E, antes de começarmos: *não tenha medo de experimentar*. O sistema não morde.Você não pode destruir alguma coisa ao trabalhar no sistema. O Linux tem segurança embutida que previne usuários "normais" de danificarem arquivos essenciais ao sistema. Mesmo assim, a pior coisa que pode acontecer é que você apague alguns ou todos os seus arquivos e tenham que reinstalar o sistema. Assim, por agora você não tem nada a peder.

2.2 Conceitos Básicos do Linux

O Linux é um sistema operacional multi-tarefa e multi-usuário, o que significa que muitas pessoas podem rodar aplicações diferentes em um computador ao mesmo tempo. Isso difere do MS-DOS, por exemplo, onde somente uma pessoa pode usar o sistema de cada vez. No Linux, para se identificar, você deve "logar" no sistema, o que exige que você forneça sua identificação (login - o nome que o sistema usa para identificá-lo), e sua senha, que é sua chave pessoal para entrar na sua conta. Como somente você sabe sua senha, ninguém pode "logar" no sistema usando seu nome de usuário.

Em sistemas UNIX tradicionais, o administrador do sistema determina um nome de usuário e uma senha inicial quando você recebe uma conta do sistema. Entretanto, como neste Linux você é o administrador do sistema, você deve configurar sua conta pessoal antes de poder logar. Para os próximos debates usaremos o nome de usuário imaginário "larry".

Além disso, cada sistema tem um nome de máquina (host name) determinado. É o hostname que dá nome a sua máquina; dá a ela caráter e charme. O hostname é usado para identificar máquinas individuais na rede, mas mesmo que sua máquina não esteja em rede, ela deve ter um hostname. Para o nosso exemplo, o nome da máquina será "mousehouse".

2.2.1 Criando uma conta

Antes que possa usar um sistema Linux recém instalado, você deve criar uma conta própria. Não é uma boa idéia usar a conta root para uso normal; você deve reservá-la para rodar comandos privilegiados e para manutenção do sistema, como discutido abaixo.

Para criar sua conta, entre como root e use o comando useradd ou adduser.

2.2.2 Logando

No momento do login, você vai ver um prompt parecido com:

mousehouse login:

Entre seu nome de usuário e pressione <Enter>. Nosso herói, larry, iria digitar:

mousehouse login: larry

Password:

A seguir, entre sua senha. Os caracteres que você entrar não serão mostrados na tela, por isso digite-os com cuidado. Se você errar sua senha, vai ver a mensagem "Login incorrect" e deve tentar novamente.

Uma vez que tenha entrado com sucesso o nome de usuário e senha, você entrou oficialmente no sistema e está livre para perambular.

2.2.3 Consoles Virtuais.

O **console** do sistema é o monitor e teclado que estão diretamente ligados no sistema. Como o Linux é um sistema operacional multi-usuário, você pode ter outros terminais conectados a portas seriais no seu sistema, mas esses não seriam o console. O Linux, como outras versões do UNIX, fornece acesso a **consoles virtuais** (ou VCs), que permitem que você tenha mais de uma sessão de login no console ao mesmo tempo.

Para demonstrar isso, entre no sistema. Após, pressione Alt-F2. Você deve ver o prompt "login:" novamente. Você está na frente do segundo console virtual. Para mudar novamente para o primeiro VC, pressione Alt-F1. *Voila!* Você está novamente na primeira sessão de login.

Um sistema Linux novo provavelmente permite que você acesse somente a primeira meia dúzia de VCs, pressionando Alt-F1 até Alt-F6, ou conforme o número de VCs configurados no sistema. É possível habilitar até 12 VCs - um para cada tecla de função no seu teclado. Como você pode ver, o uso de VCs pode ser bem poderoso, pois você pode trabalhar em várias sessões diferentes ao mesmo tempo.

Mesmo que o uso de VCs seja um tanto quanto limitado (afinal, você pode ver somente um VC de cada vez), deve dar uma noção das capacidades multi-usuários do Linux. Enquanto estiver trabalhando com o primeiro VC, você pode mudar para o segundo VC e trabalhar em alguma outra coisa.

2.2.4 Shells e comandos

A maioria de sua exploração no mundo Linux se dará através de um **shell**, um programa que pega comandos que você digita e os traduz em instruções para o sistema operacional. Ele pode ser comparado ao COMMAND.COM, o programa que no MS-DOS faz essencialmente a mesma coisa. Um shell é só uma interface para o Linux. Há muitas interfaces possíveis - como o Sistema X Window, que permite que você rode comandos usando o mouse e teclado.

Assim que você entra no login, o sistema inicia um shell e você pode começar a digitar comandos. Aqui vemos um exemplo rápido. O Larry "loga" e fica esperando no prompt do shell:

mousehouse login: larry Password: (não mostrada) Welcome to Mousehouse! /home/larry#

A última linha desse texto é o prompt do shell, indicando que está pronto para receber comandos. (Adiante veremos mais sobre o que ele significa). Vamos tentar dizer ao sistema para fazer algo interessante:

/home/larry# make love make: *** No way to make target 'love'. Stop. /home/larry#

Bem, como vemos, make é o nome de um programa real no sistema, e o shell executou o programa quando demos o comando. (Infelizmente, o sistema não foi muito amigável.)

Isto nos leva a uma pergunta picante: O que é um comando? O que acontece quando você digita "make love"? A primeira palavra na linha de comando, "make", é o nome de um comando a ser executado. Todo o reste na linha de comando é tomado como parâmetro para o comando. Exemplo:

/home/larry# cp foo bar

O nome deste comando é "cp", e os parâmetros são "foo" e "bar".

Quando você entra um comando, o shell faz várias coisas. Primeiro, ele verifica o comando para ver se ele é interno ao shell. Isto é, um comando que o próprio shell saiba como executar. Há vários comandos assim, e vamos passar por eles mais tarde. O shell também verifica se o comando não é um alias (um nome substituto) para outro comando. Se nenhuma dessas condições se aplicar, o shell procura um program no disco que tenha o nome especificado. Se for bem sucedido, o shell roda esse programa, enviando os parâmetros especificados na linha de comando.

No nosso exemplo, o shell procura por um programa chamado "make", e roda ele com o parâmetro "love". "Make" é um programa freqüentemente usado para compilar programas grandes, e recebe como parâmetros o nome do "alvo" a ser compilado. No caso do "make love", nós instruímos o "make" a compilar o arquivo "love". Como o "make" não pôde encontrar um destino com

esse nome, ele falhou com uma mensagem de erro engraçada e nos retornou para o prompt do shell.

O que acontece se digitarmos um comando para o shell e o shell não conseguir encontrar um programa com o nome especificado? Bem, podemos tentar o seguinte:

/home/larry# coma lixo coma: command not found /home/larry#

Bem simples, se o shell não pode encontrar o comando com o nome dado (no exemplo, "coma"), ele imprime uma mensagem de erro. Você vai ver esta mensagem de erro com freqüência se digitar um comando errado (ex: se você digitou "mkae love" ao invés de "make love").

2.2.5 Logout

Antes de nos aprofundarmos ainda mais, temos que lhe contar como dar logout (sair) do sistema. No prompt do shell, use o comando "exit" para dar logout. Há outras formas ainda, mas esta é a que envolve menos riscos.

2.2.6 Mudando sua senha

Você deve saber como mudar sua senha no sistema. O comando "passwd" pede pela sua velha senha e por uma nova. Ele também pede que você redigite a nova para validação. Cuidado para não esquecer sua senha - se você esquecê-la, só o administrador do sistema pode recuperar seu acesso.

2.2.7 Arquivos e diretórios

Na maioria dos sistemas operacionais (incluindo o Linux), há o conceito de **arquivo**, que é um monte de informações com um nome (chamado **filename** - nome de arquivo). Exemplos de arquivos são seu trabalho de conclusão de curso, uma mensagem de e-mail, ou um programa que pode ser executado. Essencialmente, qualquer coisa gravada no disco é gravada em um arquivo individual.

Arquivos são identificados por seus nomes. Por exemplo, um arquivo contendo seu trabalho de conclusão pode ser gravado como nome *trabalho-de-conclusão*. Esses nomes usualmente identificam o arquivo e seu conteúdo de alguma forma que faça sentido para você. Não há um formato padrão para nomes de arquivo como existe no MS-DOS ou Windows e outros sistemas operacionais. Em geral, um nome de arquivo pode conter um caracter (exceto o caracter /) e é limitado a 256 caracteres de comprimento.

Com o conceito de arquivos, vem o conceito de diretórios. Um **diretório** é uma coleção de arquivos. Você pode pensar nele como uma "pasta" que contém muitos arquivos diferentes. Diretórios também recebem nomes, com os quais você os identifica. Além disto, diretórios são mantidos em uma estrutura tipo árvore. Isso significa que diretórios podem conter outros diretórios.

Conseqüentemente, você pode se referenciar a um arquivo pelo seu caminho (path), que é composto do nome de arquivo, precedido pelo nome do diretório que contém o arquivo. Vamos assumir que o Larry, por exemplo, tenha um diretório chamado "papers", que contém três arquivos: history-final, english-lit, e tese-mestrado. Cada um desses três arquivos contém informação sobre um dos projetos do Larry. Para se referir ao arquivo english-lit, o Larry pode especificar o path como "papers/english-lit".

Como você pode ver, o diretório e nome são separados por uma barra simples (/). Por este motivo, os nomes de arquivo não podem conter o caracter /. Os usuários do MS-DOS irão achar esta convenção familiar, embora no MS-DOS seja usada a contra-barra (\).

Como mencionado, diretórios podem ser aninhados dentro de outros. Vamos assumir, por exemplo, que exista outro diretório dentro de "papers", chamado "notas". O diretório "notas" contém os arquivos notas-matematica e notas-rascunho. O path do arquivo notas-rascunho será:

papers/notas/notas-rascunho

Assim, o path é como um caminho para o arquivo. O diretório que contém um dado subdiretório é conhecido como diretório superior (parent directory). Aqui, o diretório papers é superior ao diretório notas.

2.2.8 A árvore de diretórios

A maioria dos sistemas Linux usa um formato padrão para guardar os arquivos, de forma a que recursos e programas possam ser facilmente localizados. Esse formato forma a árvore de diretórios, que começa com o diretório "/", também chamado de diretório raiz. Diretamente abaixo do "/" estão importantes subdiretórios: /bin, /etc, /dev e /usr, dentre outros. Esses diretórios, por sua vez, contêm outros diretórios que contém arquivos de configuração do sistema, programas, e assim por diante.

Cada usuário tem um **diretório home** (**home directory**), que é o diretório reservado para que aquele usuário armazend seus arquivos. Nos exemplos acima, todos os arquivos do Larry (como notas-rascunho e tesemestrado) estão contidos no diretório home do Larry. Usualmente, diretórios home estão abaixo do diretório /home, e seus nomes refletem os usuários aos quais os diretórios pertencem. O diretório home do Larry, por exemplo, é /home/larry.

O diagrama abaixo mostra um exemplo de árvore de diretórios que pode lhe dar uma idéia de como a árvore de diretórios do seu sistema está organizada.

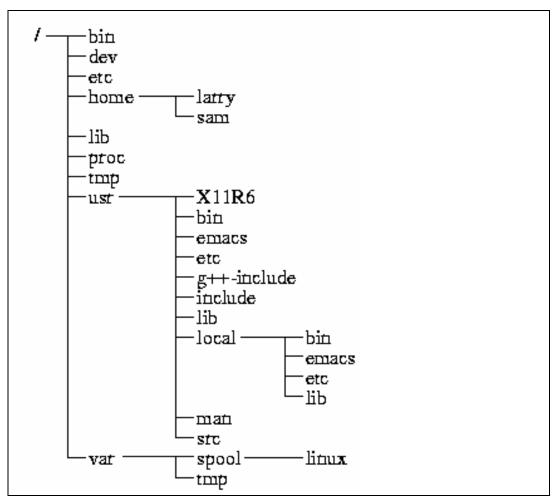


Figura 3.1: Uma árvore de diretó rios típica (abreviada) do Linux.

2.2.9 O diretório de trabalho corrente

A qualquer momento, assume-se que comandos que você digita são relativos ao seu **diretório de trabalho corrente**. Você pode pensar no seu diretório de trabalho como o diretório onde você está atualmente localizado. Quando você entra no sistema, seu diretório de trabalho é configurado como seu diretório home - /home/larry, no seu caso. Sempre que se referir a um arquivo, você pode fazê-lo em relação ao seu diretório de trabalho corrente, ao invés de ter que ficar especificando o caminho do arquivo (path).

Aqui vemos um exemplo. Larry tem o diretório papers, e papers contém o arquivo history-final. Se o Larry quer ver este arquivo, ele pode usar o comando

/home/larry# more /home/larry/papers/history-final

O comando "more" simplesmente mostra o arquivo na tela, uma página por vez. Entretanto, como o diretório de trabalho corrente do Larry é /home/larry, ele pode se referenciar ao arquivo de forma relativa a sua localização atual usando o comando

/home/larry# more papers/history-final

Se você começa um nome de arquivo com o caracter diferente de / (como em papers/final), você está se referenciado a este arquivo de forma relativa ao seu diretório de trabalho corrente. Chamamos isso de **caminho relativo** (**relative path name**).

Por outro lado, se você começa um nome de arquivo com uma "/", o sistema interpreta isso como o caminho completo - isto é, um caminho que inclui o caminho completo para o arquivo, começando com o diretório raiz, /. Isso é chamado de **caminho absoluto** (absolute path name)

2.2.10 Referindo-se a diretórios home

Tanto com o tcsh quanto com o bash, você pode especificar o diretório home com um caracter til (~). Por exemplo, o comando:

/home/larry# more ~/papers/history-final

é equivalente a

/home/larry# more /home/larry/papers/history-final

O shell substitui o caracter ~ pelo nome do seu diretório home.

Você também pode especificar diretórios home de outros usuários com o caracter ~. O caminho "~karl/letters" é traduzido para "/home/karl/letters" pelo shell (se /home/karl for o diretório home do karl). Usar o til é como usar um atalho. Não há um diretório chamado "~" - é só uma mãozinha oferecida pelo shell.

2.3 Primeiros passos no Linux

Antes de começarmos, é importante ter em mente que todos os nomes de arquivos e comandos no Linux fazem diferenciação de letras maiúsculas e minúsculas (diferentemente do Windows ou MS-DOS). O comando "make", por exemplo, é diferente de "Make" ou "MAKE". O mesmo vale para nomes de arquivos e diretórios.

2.3.1 Passeando por aí

Agora que você pode logar e sabe se referir a arquivos usando caminhos, como pode mudar o diretório de trabalho corrente para tornar a vida mais fácil?

O comando para mover-se por aí na estrutura de diretórios é "cd", que é um nome curto para "change directory", ou "mudar de diretório". Muitos dos comandos mais usados do Linux têm duas ou três letras. O uso do comando cd é:

cd diretório

onde diretório é o nome do diretório que você deseja que se torne seu diretório de trabalho corrente. Se o Larry quisesse mudar para o subdiretório papers, ele usaria o comando:

/home/larry# cd papers /home/larry/papers#

Como você pode ver, o prompt do Larry mudou para refletir seu diretório de trabalho corrente (assim ele sabe onde está). Agora que ele está no diretório papers, ele pode ver o arquivo history-final com o comando:

/home/larry/papers# more history-final

Agora o Larry está trancado no subdiretório papers. Para mover-se para cima para o diretório imediatamente superior (ou parent), use o comando:

/home/larry/papers# cd ..

/home/larry#

(Note o espaço entre o "cd" e o "..").

Cada diretório tem uma entrada chamada ".." que se refere ao diretório superior. Similarmente, cada diretório tem uma entrada chamada ".", que se refere a ele mesmo. Assim, o comando:

/home/larry/papers# cd .

nos leva a lugar nenhum.

Você também pode usar caminhos absolutos no comando cd. Para ir para o diretório home do Karl, podemos usar o comando:

/home/larry/papers# cd /home/karl

/home/karl#

Usando somente cd, sem parâmetro algum, irá retorná-lo ao seu diretório home:

/home/karl# cd /home/larry#

2.3.2 Visualizando o conteúdo de diretórios

Agora que você sabe como se mover pelos diretórios, você pode pensar: "E aí?" Mover-se pelos diretórios por si só é inútil. Vamos introduzir, então, um novo comando: o "Is". O comando "Is" mostra uma lista de arquivos e diretórios - por default do seu diretório corrente. Por exemplo:

/home/larry# ls

Mail

letters

papers

/home/larry#

Aqui podemos ver que o Larry tem três entradas no seu diretório corrente: Mail, letters e papers. Isso não nos diz muito - eles são arquivos ou diretórios? Podemos usar a opção -F do comando ls para ver informação mais detalhada.

```
/home/larry# Is -F
Mail/
letters/
papers/
/home/larry#
```

Através da barra (/) adicionada a cada nome de arquivo, sabemos que estas três entradas são de fato subdiretórios.

O "Is -F" pode também acrescentar um asterisco (*) no fim dos arquivos, o que indicaria que o arquivo é **executável**, ou um programa que pode ser rodado. Se nada for acrescentado ao nome do arquivo usando "Is -F", o arquivo é um "arquivo simples" (plain), isto é, nem um diretório nem um executável.

Em geral, cada comando UNIX pode receber um número de opções, além dos outros parâmetros. Essas opções usualmente começam com um "-", como demonstrado acima com a opção "-F". A opção "-F" diz ao Is para dar mais informações sobre o tipo dos arquivos envolvidos - neste cado, imprimindo uma "/" depois do nome de cada diretório.

Se você fornecer ao ls um nome de diretório, o sistema irá mostrar o conteúdo desse diretório.

```
/home/larry# Is -F papers
english-lit
history-final
tese-mestrado
notas/
/home/larry#
```

Ou, para uma listagem mais interessante, vamos ver o que há no diretório /etc do sistema:

/home/larry#	ls /etc			
Images	ftpusers	lpc	rc.new	shells
adm	getty	magic	rc0.d	startcons
bcheckrc	gettydefs	motd	rc1.d	swapoff
brc	group	mount	rc2.d	swapon
brc~	inet	mtab	rc3.d	syslog.conf
csh.cshrc	init	mtools	rc4.d	syslog.pid

Se você for um usuário MS-DOS, você pode notar que os nomes de arquivos podem ser maiores que 8 caracteres, e que podem conter pontos em qualquer posição. Você pode até usar mais de um ponto num nome de arquivo.

Vamos nos mover para o topo da árvore de diretórios, e depois para baixo, para outro diretório, com os comandos:

```
/home/larry# cd ..
/home# cd ..
/# cd usr
/usr# cd bin
/usr/bin#
```

Você também pode se mover para dentro de vários diretórios em um único passo, como em "cd /usr/bin".

Tente movimentar-se por vários diretórios, usando o ls e o cd. Talvez você encontre algumas mensagens do tipo "Permission denied". Trata-se apenas da segurança do UNIX mostrando suas garras: para usar o ls ou cd, você deve ter permissão para isso.

2.3.3 Criando novos diretórios

É hora de aprender a criar diretórios. Isso envolve o uso do comando mkdri. Tente o seguinte:

```
/home/larry# mkdir foo
/home/larry# ls -F
Mail/
foo/
letters/
papers/
/home/larry# cd foo
/home/larry/foo# ls
/home/larry/foo#
```

Parabéns! Você criou um novo diretório e entrou nele. Como não há arquivos nesse novo diretório, vamos aprender como copiar arquivos de um lugar para o outro.

2.3.4 Copiando arquivos

```
Para copiar arquvis, use o comando cp, como mostrado:
/home/larry/foo# cp /etc/termcap .
/home/larry/foo# cp /etc/shells .
/home/larry/foo# ls -F
shells termcap
/home/larry/foo# cp shells bells
/home/larry/foo# ls -F
```

bells shells termcap /home/larry/foo#

O comando cp copia os arquivos listados na linha de comando para o arquivo ou diretório dado como último argumento. Note que usamos um ponto (".") para nos referirmos ao diretório corrente.

2.3.5 Movendo arquivos

O comando mv move arquivos, ao invés de copiá-los. A sintaxe é bem direta:

/home/larry/foo# mv termcap sells /home/larry/foo# ls -F bells sells shells /home/larry/foo#

Note que o arquivo termcap foi renomeado para sells. Você também pode usar o comando mv para mover um arquivo para um diretório completamente novo.

Nota: o mv e o cp vão sobreescrever o arquivo destino que tiver o mesmo nome sem pedir por confirmação. Tome cuidado quando mover um arquivo em outro diretório. Pode já existir um nome igual no mesmo diretório, que você irá sobreescrever!

2.3.6 Excluindo arquivos e diretórios

Para excluir um arquivo, use o comando "rm" (que vem de "remover"), como mostrado aqui:

/home/larry/foo# rm bells sells /home/larry/foo# ls -F shells /home/larry/foo#

Ficamos sem nada, a não ser "shells", mas não nos queixamos. Note que o rm, por default, não vai pedir confirmação antes de apagar um arquivo - tome cuidado! Note também que você pode apagar vários arquivos, passando uma lista como parâmetro ao "rm".

Um comando relacionado ao rm é o rmdir. Esse comando exclui um diretório, mas somente se o diretório estiver vazio. Se o diretório contém qualquer arquivo, ou subdiretório, o rmdir vai reclamar.

2.3.7 Visualizando arquivos

Os comandos "more" e "cat" são usados para visualizar o conteúdo de arquivos. O "more" mostra um arquivo, uma tela por vez, enquanto o "cat" mostra todo o arquivo de uma vez.

Para ver o arquivo shells, use o comando: /home/larry/foo# more /etc/shells

Esse arquivo tem uma lista dos programas shell disponíveis no seu sistema (geralmente está no /etc). Na maioria dos sistemas, ele inclui o /bin/sh, o /bin/bash e o /bin/csh. Vamos falar sobre os diferentes tipos de shell mais tarde.

Enquanto estiver usando o "more", pressione a tecla de espaços para mostrar a próxima página, e "b" para mostrar a anterior. Há outros comandos disponíveis no "more". Estes são somente os básicos. Pressione "q" para sair.

Saia do "more" e tente dar um cat em /etc/services. O texto provavelmente irá passar tão rapidamente que você não poderá lê-lo. O nome "cat" vem de "concatenar", que é para o que serve esse programa. O comando "cat" pode ser usado para concatenar o conteúdo de vários arquivos e salvar o resultado em outro arquivo.

2.3.8 Buscando ajuda online

Quase todo o sistema UNIX, incluindo o Linux, fornece uma facilidade chamada de **páginas do manual** (**manual pages**). O manual contém documentação online para os comandos de sistema, recursos, arquivos de configuração e assim por diante.

O comando usado para acessar páginas do manual é o man. Se você está interessado em aprender sobre outras opções do comando "ls", por exemplo, pode digitar:

/home/larry# man Is

e a página do manual para o ls vai ser mostrada.

Infelizmente, muitas páginas de manual são escritas para aqueles que já tem alguma idéia do que o comando ou recurso faz. Por este motivo, as páginas usualmente contêm somente detalhes técnicos do comando, sem muita explicação. Entretanto, este pode ser um recurso inestimável para refrescar sua memória se você esqueceu a sintaxe do comando. As páginas do manual também vão contar-lhe sobre comandos que nós não iremos cobrir neste livro.

Eu sugiro que você tente o man para comandos que nós já experimentamos antes e sempre que eu introduzir um novo comando. Alguns não terão um manual, por várias razões. Primeiro, a página pode ainda não ter sido escrita. (O Linux Documentation Project é responsável pelas páginas de manual no Linux). Segundo, o comando pode ser um comando interno do shell ou um alias, que não teria uma página de manual própria. Um exemplo é o cd, que é um comando interno do shell. O shell mesmo processa o cd - não há um programa separado que implementa esse comando.

2.4 Comandos básicos do UNIX

Esta seção introduz alguns dos comandos básicos mais úteis num sistema UNIX, incluindo aqueles que foram discutidos nas seções anteriores.

Note que opções usualmente começam com "-", e na maioria dos casos você pode especificar mais de uma opção em um único "-". Por exemplo, ao invés de usar o comando "ls -I -F", você pode usar "ls -IF".

Ao invés de listar todas as opções para cada comando, vamos apresentar somente os comandos úteis ou importantes por agora. De fato, a maioria dos comandos tem muitas opções que você nunca vai usar. Você pode usar o man para ver as páginas de manual para cada comando, que listam todas as opções disponíveis.

Note também que muitos desses comandos têm como parâmetros uma lista de arquivos ou diretórios, marcados nesta tabela como "arquivo1 ... arquivoN". Por exemplo, o comando cp recebe como parâmetro uma lista de arquivos para copiar, seguida de um arquivo ou diretório destino. Quando estiver copiando mais de um arquivo, o destino deve ser um diretório.

3 COMANDOS PARA MANIPULAÇÃO DE DIRETÓRIOS

Abaixo comandos úteis para a manipulação de diretórios. Um diretório é usado para armazenar arquivos de um determinado tipo. O diretório pode ser entendido como uma *pasta* onde você guarda seus papeis (arquivos). Como uma pessoa organizada, você utilizará uma pasta para guardar cada tipo de documento, da mesma forma você pode criar um diretório vendas para guardar seus arquivos relacionados com vendas naquele local.

3.1 O comando 1s

Mostra informação sobre os nomes de arquivos e diretórios.

Sintaxe:

ls [opções] [caminho/arquivo] [caminho1/arquivo1] ... onde:

- caminho/arquivo: Diretório/arquivo que será listado.
- caminho1/arquivo1: Outro Diretório/arquivo que será listado. Podem ser feitas várias listagens de uma só vez.
- opções: Modificam o comportamento do comando
 - -a Lista todos os arquivos (inclusive os ocultos) de um diretório.
 - -1 Usa o formato longo para listagem de arquivos.
 Lista as permissões, data de modificação, donos, grupos, etc.
 - F Insere um caracter após arquivos executáveis ('*'), diretórios ('/'), soquete ('='), link simbólico ('@') e pipe ('|'). Seu uso é útil para identificar de forma fácil tipos de arquivos nas listagens de diretórios.
 - --color=PARAM Mostra os arquivos em cores diferentes, conforme o tipo de arquivo. PARAM pode ser:
 - § never nunca lista em cores (mesma coisa de não usar o parâmetro --color).
 - § always sempre lista em cores conforme o tipo de arquivo.
 - § auto somente colore a listagem se estiver em um terminal.

A listagem pode ser classificada usando-se as seguintes opções:

- -f Não ordena, e usa -au para listar os arquivos.
- -r Inverte a seqüência de ordenação.
- -c Ordena pela data de alteração.
- -x Ordena pela extensão.
- –U Não ordena, lista os arquivos na ordem do diretório.

Exemplos:

lista os arquivos do diretório atual.

ls /bin /sbin lista os arquivos do diretório /bin e /sbin

ls -la /bin listagem detalhada e completa dos arquivos do diretório

/bin

3.2 O comando cd

Muda o diretório de trabalho atual. É necessário ter a permissão de execução para entrar no diretório.

Sintaxe:

cd [diretorio]

onde:

diretorio - diretório para onde se deseja mudar. "." se refere ao diretório corrente, ".." ao diretório superior na hierarquia, e "~" é o diretório padrão do usuário. Se nenhum diretório for especificado, "~" é assumido.

Exemplos:

cd retorna ao diretório padrão do usuário.

cd / muda para o diretório raiz.

cd /etc muda para o diretório /etc.

cd - retorna ao diretório anteriormente acessado.

cd .. sobe um diretório na hierarquia.

3.3 O comando pwd

O comando pwd pode ser usado para verificar em qual diretório o usuário se encontra, caso seu aviso de comandos não mostre isso.

Exemplo:

pwd mostra o nome e caminho do diretório atual

3.4 O comando mkdir

Cria um novo diretório.

Sintaxe:

- caminho Caminho onde o diretório será criado.
- diretório Nome do diretório que será criado.
- opções: Modificam o comportamento do comando:

o --verbose mostra uma mensagem para cada diretório criado. As mensagens de erro serão mostradas mesmo que esta opção não seja usada.

Para criar um novo diretório, você deve ter permissão de gravação no diretório pai do novo diretório.

Exemplos:

```
mkdir /home/larry/test cria o diretório test em /home/larry.
mkdir test1 test2 cria os diretórios test1 e test2 dentro do diretório atual.
```

3.5 O comando rmdir

Remove diretórios. Este comando faz exatamente o contrário do mkdir. O diretório a ser removido deve estar vazio e você deve ter permissão de gravação para remove-lo. Além disso, o diretório de trabalho atual não deve estar dentro do diretório a ser excluído.

Sintaxe:

- caminho Caminho do diretório que será removido.
- diretório Nome do diretório que será removido.

Para remover diretórios que contenham arquivos, use o comando rm com a opção -r (para maiores detalhes, veja rm, Seção 9.3).

Exemplos:

rmdir /home/larry/test remove o diretório test em /home/larry.
rmdir test1 test2 remove os diretórios test1 e test2 de
dentro do diretório atual.

4 EXPLORANDO O SISTEMA DE ARQUIVOS

Um sistema de arquivos é a coleção de arquivos e a hierarquia de diretórios em um sistema. Chegou a horá de guiá-lo pelo sistema de arquivos.

Você já tem as habilidades e conhecimento para entender o sistema de arquivos do Linux, e tem um mapa (veja a figura 3.2.8).

Primeiro, mude para o diretório root ("cd /"), e entre "ls -F" para mostrar uma lista do seu conteúdo. Provavelmente você vai ver os seuintes diretórios: bin, dev, etc, home, install, lib, mnt, proc, root, tmp, user, usr, e var.

Vamos dar uma olhada em cada um deles.

/bin

/bin é uma abreviatura para "binários", ou executáveis. É onde residem a maioria dos programas essencias do sistema. Use o comando "Is -F /bin" para listar os arquivos. Poderá ver alguns arquivos que reconhecerá, como cp, Is e mv. Esses são os arquivos que contêm os programas para esses comandos. Quando você executa cp, está executando o programa /bin/cp.

Usando "Is -F" verá que a maioria (se não todos) dos arquivos em /bin têm um asterisco ("*") acrescrentado ao final de seus nomes. Isso indica que são arquivos executáveis, como descrito na seção 3.3.2.

/dev

A seguir vem o /dev. Vamos dar uma olhada dentro dele, com "ls -F /dev".

Os arquivos em /dev são conhecidos como controladores de dispositivo (device drivers) e são usados para acessar os dispositivos ou recursos do sistema, como discos rígidos, modems, memória, etc. Por exemplo, da mesma forma que você pode lear dados de um arquivo, pode também ler da entrada do mouse, lendo /dev/mouse.

Os arquivos que começam com fd são os controladores de disquete. fd0 é o primeiro drive, e fd1 o segundo. Agora, se você for esperto, se dará conta que há mais controladores de dispositivo para drivers do que mencionamos. Eles representam tipos específicos de discos. Por exemplo, fd1H1440 acessa discos de 3.5" de alta densidade no drive 1.

Aqui temos uma lista de alguns dos controladores de dispositivos mais usados. Note-se que mesmo que você não tenha algum dos dispositivos listados, terá entradas em dev de qualquer forma.

- /dev/console faz referência ao console do sistema, quer dizer, ao monitor conectado em seu sistema.
- Os dispositivos /dev/ttyS e /dev/cua são usados para acessar as portas seriais. Por exemplo, /dev/ttyS0 faz referência a "COM1", sob o MS-DOS. Os dispositivos /dev/cua são "callout" e são usados em conjunto com um modem.
- Os nomes dos dispositivos que começam por hd acessam a discos rígidos. /dev/hda se refere a todo o primeiro disco, enquanto que /dev/hda1 se refere a primeira partição em /dev/hda.
- Os nomes de dispositivo que començam com sd são dispositivos SCSI. Se você tem um disco rígido SCSI, no lugar de acessá-lo

- com /dev/hda, vai acessá-lo com /dev/sda. As fitas SCSI são acessadas via dispositivos st e os CD-ROM SCSI via sr.
- Os nomes que començam por lp acessam as portas paralelas. /dev/lp0 se refere a "LPT1" no mundo MS-DOS.
- /dev/null é usado como "buraco negro", qualquer dado enviado a este dispositivo desaparece. Para que pode ser útil isto?. Bem, se deseja suprimir a saída para a tela de um comando, pode enviar a saída para /dev/null. Falaremos mais sobre isto depois.
- Os nomes que começam por /dev/tty se referem a "consoles virtuais" de seu sistema (acessíveis mediante as teclas Alt-F1, Alt-F2, etc).
- /dev/tty1 se refere a primeira VC, /dev/tty2 a segunda, etc.
- Os nomes de dispositivo que começam com /dev/pty são "pseudoterminais". São usados para proporcionar um "terminal" a sessões remotas. Por exemplo, se sua máquina está em uma rede, o telnet de entrada usará um dos dispositivos /dev/pty.

/etc

/etc contém uma série de arquivos de configuração do sistema. Isto inclui o /etc/passwd (a base de dados de usuários), o /etc/rc (instruções de inicialização do sistema), etc.

/sbin

/sbin é usado para armazenar programas esenciáis do sistema, usados pelo administrador do sistema.

/home

/home contém os diretórios "home" dos usuários. Por exemplo, /home/larry é o diretório do usuário "larry". Em um sistema recém instalado, esse diretório estará vazio.

/lib

/lib contém as imagens das bibliotecas compartilhadas. Esses arquivos contêm código que muitos programas compartilham. Ao invés de cada programa ter uma cópia própria das funções compartilhadas, elas são guardadas em um lugar comum, o /lib. Isso faz com que os programas executáveis sejam menores e reduzam o espaço usado em disco.

/proc

/proc é um "sistema de arquivos virtual". Os arquivos que estão no /proc residem verdadeiramente na memória, e não no disco. Esses arquivos se referem a vários processos que rodam no sistema e permitem obter informação sobre programas e processos que estão rodando num dado momento. Falaremos mais sobre isso na seção 3.11.1.

/tmp

Muitos programas têm necessidade de gerar alguma informação temporária e de guardar essa informação em um arquivo temporário. O lugar para esses arquivos é o /tmp.

/usr

/usr é um diretório muito importante. Contém uma série de subdiretórios que por sua vez contêm alguns dos mais importantes e úteis programas e arquivos de configuração usados no sistema.

Os vários diretórios descritos acima são essenciais para o sistema operar, mas a maioria dos itens no /usr são opcionais. Entretanto, são esses itens opcionais que tornam o sistema útil e interessante. Sem o /usr, você teria um sistema tedioso que suportaria somente programas como cp e ls. O /usr contém a maioria dos pacotes grandes de software e os arquivos de configuração que os acompanham.

/usr/X11R6

/usr/X11R6 contém o sistema X Window se você o instalou el sistema X Window se você o instalou. O sistema X Window é um ambiente gráfico poderoso que proporciona um grande número de ferramentas e programas gráficos, mostrados em janelas na sua tela. Se você está familiarizado com os ambientes MS Windows ou Macintosh, o X Window lhe será muito familiar. O diretório /usr/X11R6 contém todos os executáveis do X Window, arquivos de configuração e suporte. Entraremos em mais detalhes sobre isso na seção 5.1.

/usr/bin

/usr/bin contém a maioria dos programas executáveis não encontrados em outras partes, como no /bin.

/usr/etc

Como o /etc, contém diferentes arquivos de configuração e programas do sistema, /usr/etc contém inclusive mais que o anterior. Em geral, os arquivos que se encontram em /usr/etc não são essenciais para o sistema, diferentemente dos que se encontram no /etc.

/usr/include

/usr/include contém os arquivos *include* para o compilador C. Esse arquivos (a maioria dos quais termina com .h, de "header") declaram estruturas de dados, subrotinas e constantes usadas no desenvolvimento de programas em C. Os arquivos que se encontram em /usr/include/sys geralmente são usados quando programando no nível do sistema UNIX. Se você está familiarizado com programação C, encontrará arquivos como stdio.h, o qual declara funções como printf().

/usr/q++-include

/usr/g++-include contém arquivos de inclusão para o computador C++ (muito parecido ao /usr/include).

/usr/lib

/usr/lib contém as bibliotecas "stub" e "static" equivalentes aos arquivos encontrados em /lib. Ao compilar um programa, ele é "linkado" com as bibliotecas que se encontram em /usr/lib, as quais direcionam o programa para o /lib, quando precisa buscar o código real da biblioteca. Além disso, vários programas armazenam arquivos de configuração no /usr/lib.

/usr/local

/usr/local é muito parecido ao /usr. Ele contém programas e arquivos não essenciais ao sistema, mas que tornam o sistema mais divertido e excitante. Em geral, os programas que se encontram em /usr/local são específicos do seu sistema, isto é, o diretório /ussr/local difere bastante entre os sistemas UNIX. Aqui encontrará grandes programas, como o TeX (sistema de formatação de documentos) e Emacs (editor grande e poderoso), se os instalar.

/usr/man

Esse diretório contém as páginas de manual. Há dois subdiretórios para cada seção de página de manual (use o comando "man man" para detalhes). Por exemplo, /usr/man/man1 contém os fontes (isto é, o original não formatado) para as páginas de manual na seção 1, e /usr/man/cat1 contém as páginas de manual formatadas para a seção 1.

/usr/src

/usr/src contém o código fonte (programas a compilar) de vários programas do sistema. O subdiretório mais importante é o /usr/src/linux, que contém o código fonte do kernel do Linux.

/var

/var contém diretórios que freqëntemente mudam de tamanho ou tendem a crescer. Muitos desses diretórios residiam antes em /usr, mas desde que estamos tratando de deixar o /usr inalterado, os diretórios que mudam de tamanho foram levados para o /var. Alguns deles são:

/var/adm

/var/adm contém vários arquivos de interesse para o administrador do sistema, especificamente históricos do sistema, que armazenam erros ou problemas com o sistema. Outros arquivos guardam logins do sistema, assim como tentativas frustradas. O capítulo 4 abordará este assunto.

/var/spool

/var/spool contém arquivos que vão ser passados a outros programas. Por exemplo, se sua máquina está conectada a uma rede, o correio de entrada será armazenado em /var/spool/mail até que você o leia ou apague. Artigos novos dos News, tanto os que entram quanto os que saem, se encontram em /var/spool/news, etc.

5 COMANDOS PARA MANIPULAÇÃO DE ARQUIVOS

Abaixo, comandos utilizados para manipulação de arquivos.

5.1 O comando rm

Remove arquivos. Também pode ser usado para apagar diretórios, vazios ou não. Note que no UNIX quando um arquivo é removido, ele não é mais recuperável (como no MS-DOS, onde você pode usar o "undelete", ou no Windows, com a lixeira).

Sintaxe:

rm [opções] [caminho][arquivo/diretório]

onde:

- caminho: Localização do arquivo que deseja apagar. Se omitido, assume que o arquivo esteja no diretório atual.
- arquivo/diretório: Arquivo/diretório que será apagado.
- *opções:* Modificam o comportamento do comando:
 - o -i, --interactiveConfirma antes de remover, esta é ativada por padrão.
 - o -f, --force Remove os arquivos sem confirmação.
 - o -r, --recursive Usado para remover arquivos em subdiretórios. Esta opção também pode ser usada para remover sub-diretórios.

Exemplos:

rm teste.txt Remove o arquivo teste.txt do diretório atual.

rm *.txt Remove todos os arquivos do diretório atual que terminam com .txt.

rm *.txt teste.novo Remove todos os arquivos do diretório atual que terminam com .txt e também o arquivo teste.novo.

rm -rf /tmp/teste/* Remove todos os arquivos e sub-diretórios do diretório /tmp/teste mas mantém o sub-diretório /tmp/teste.

rm -rf /tmp/teste Remove todos os arquivos e sub-diretórios do diretório /tmp/teste, inclusive /tmp/teste.

5.2 O comando cp

Copia arquivos.

Sintaxe:

cp [opções] [origem] [destino]

onde:

 origem Arquivo que será copiado. Podem ser especificados mais de um arquivo para ser copiado usando "Curingas" (veja <u>Curingas</u>, <u>Seção 2.12</u>).

- destino: O caminho ou nome de arquivo onde será copiado. Se o destino for um diretório, os arquivos de origem serão copiados para dentro do diretório.
- opções: Modificam o comportamento do comando:
 - o -f, --force Não confirma, substitui arquivos caso já existam.
 - o -i, --interactive Confirma antes de substituir arquivos existentes.
 - -u, --update Copia somente se o arquivo de origem é mais novo que o arquivo de destino ou quando o arquivo de destino não existe.
 - -r, Copia arquivos e sub-diretórios, com exceção de arquivos especiais FIFO e dispositivos.
 - o -R, --recursive Copia arquivos e sub-diretórios, inclusive arquivos especiais FIFO e dispositivos.
 - o -p, --preserve Preserva atributos do arquivo, se for possível.

O comando cp copia arquivos da ORIGEM para o DESTINO. Ambos origem e destino terão o mesmo conteúdo após a cópia.

Exemplos:

cp teste testel Copia o arquivo teste para testel

/tmp.

 ${\tt cp}$ * ${\tt /tmp}$ Copia todos os arquivos do diretório atual para

/tmp.

cp /bin/* . Copia todos os arquivos do diretório /bin para o

diretório atual.

cp -R /bin /tmp Copia o diretório /bin e todos os arquivos/sub-diretórios existentes para o diretório /tmp.

cp -R /bin/* /tmp Copia todos os arquivos do diretório /bin (exceto o próprio diretório /bin) e todos os arquivos/sub-diretórios existentes dentro dele para /tmp.

5.3 O comando my

Move ou renomeia arquivos e diretórios. O processo é semelhante ao do comando cp mas o arquivo de origem é apagado após o término da cópia.

Sintaxe:

mv [opções] [origem] [destino]

Onde:

- origem: Arquivo/diretório de origem.
- destino: Local onde será movido ou novo nome do arquivo/diretório.

• opções: Modificam o comportamento do comando:

o -f, --force Substitui o arquivo de destino sem perguntar.

o -i, --interactive Pergunta antes de substituir. É o padrão.

 -u, --update Move somente arquivos antigos, ou novos arquivos.

Exemplos:

mv teste teste1 Muda o nome do arquivo teste para teste1.
mv teste /tmp Move o arquivo teste do diretório atual para /tmp.

5.4 O comando ln

Cria um novo *link* para arquivos e diretórios. O *link* é o mecanismo usado para fazer referência a um arquivo ou diretório.

Sintaxe:

ln [opções] [origem] [link]

Onde:

• origem: Diretório ou arquivo de onde será feito o link.

• *link:* Nome do novo *link* que será criado.

• opções: Modificam o comportamento do comando:

- -s Cria um link simbólico. Usado para criar ligações com o arquivo/diretório de destino.
- -d Cria um hard link para diretórios. Somente o root pode usar esta opção.

Existem 2 tipos de links: simbólicos e rígidos.

- O link rígido faz referência ao mesmo i-node do arquivo original, desta forma ele será perfeitamente idêntico, inclusive nas permissões de acesso, ao arquivo original. Ambos os links passam a se referir ao mesmo arquivo, e não a cópias iguais. Não é possível fazer um link rígido para um diretório ou para arquivos que estejam em partições diferentes.
- O link simbólico cria um arquivo especial no disco (do tipo link) que tem como conteúdo o caminho para chegar até o arquivo alvo (isto pode ser verificado pelo tamanho do arquivo do link). A opção -s cria links simbólicos.

Observações:

- Se for usado o comando rm com um link, somente o link será removido.
- Se for usado o comando cp com um link, o arquivo original será copiado ao invés do link.
- Se for usado o comando mv com um link, a modificação será feita no link.

 Se for usado um comando de visualização (como o cat), o arquivo original será visualizado.

Exemplos:

- ln -s /dev/ttyS1 /dev/modem Cria o link simbólico /dev/modem para o arquivo /dev/ttyS1.
- ln -s /tmp ~/tmp Cria um link ~/tmp para o diretório /tmp.

5.5 O comando cat.

Oficialmente usado para concatenar arquivos, mas também pode ser usado para mostrar o conteúdo completo de um arquivo por vez.

Sintaxe:

cat [opções] [diretório/arquivo] [diretório1/arquivo1]

Onde

- diretório/arquivo: Localização do arquivo que se deseja visualizar o conteúdo.
- *opções:* Modificam o comportamento do comando:
 - o -n, --number Mostra o número das linhas enquanto o conteúdo do arquivo é mostrado.
 - o -s, --squeeze-blank Não mostra mais que uma linha em branco entre um parágrafo e outro.

O comando cat trabalha com arquivos texto. Use o comando zcat para ver diretamente arquivos compactados com gzip.

Exemplo:

cat /usr/doc/copyright/GPL

5.6 O comando tac

Mostra o conteúdo de um arquivo binário ou texto (como o cat) só que em ordem inversa.

Sintaxe:

tac [opções] [diretório/arquivo] [diretório1/arquivo1]

- diretório/arquivo: Localização do arquivo que se deseja visualizar o conteúdo
- *opções:* Modificam o comportamento do comando:
 - o -s [string] Usa o [string] como separador de registros.

Exemplo:

tac /usr/doc/copyright/GPL

5.7 O comando more

Permite fazer a paginação de arquivos ou da entrada padrão. O comando more pode ser usado como comando para leitura de arquivos que ocupem mais de uma tela. Quando toda a tela é ocupada, o more efetua uma pausa e permite que você pressione Enter ou espaço para continuar avançando no arquivo sendo visualizado. Para sair do more pressione q.

Sintaxe:

more [arquivo]

Onde:

• arquivo: É o arquivo que será paginado.

Para visualizar diretamente arquivos texto compactados pelo gzip (".gz") use o comando zmore.

Exemplos:

```
more /etc/passwd
cat /etc/passwd | more
```

5.8 O comando less

Permite fazer a paginação de arquivos ou da entrada padrão. O comando less pode ser usado como comando para leitura de arquivos que ocupem mais de uma tela. Quando toda a tela é ocupada, o less efetua uma pausa (semelhante ao more) e permite que você pressione Seta para Cima e Seta para Baixo ou PgUP/PgDown para fazer o rolamento da página. Para sair do less pressione q.

Sintaxe:

less [arquivo]

Onde:

• arquivo: É o arquivo que será paginado.

Para visualizar diretamente arquivos texto compactados pelo gzip (".gz") use o comando zless.

Exemplos:

```
less /etc/passwd
cat /etc/passwd | less
```

5.9 O comando head

Mostra as linhas iniciais de um arquivo texto.

Sintaxe:

head [opções] [arquivo]

Onde

- arguivo: É o arquivo que será mostrado.
- opções: Modificam o comportamento do comando:
 - o -c [numero] Mostra o [numero] de bytes do inicio do arquivo.

o -[numero] Mostra o [numero] de linhas do inicio do arquivo. Caso não seja especificado, mostra as 10 primeiras linhas.

Exemplos:

head teste.txt head -20 teste.txt

5.100 comando tail

Mostra as linhas finais de um arquivo texto.

Sintaxe:

tail [opções] [arquivo]

Onde:

- arquivo: É o arquivo que será mostrado.
- opções: Modificam o comportamento do comando:
 - c [numero] Mostra o [numero] de bytes do final do arquivo.
 - -[numero] Mostra o [numero] de linhas do final do arquivo. Caso não seja especificado, mostra as 10 últimas linhas.

Exemplos:

tail teste.txt
tail -n 20 teste.txt

5.110 comando touch

Muda a data e hora que um arquivo foi alterado. Também pode ser usado para criar arquivos vazios. Caso o touch seja usado com arquivos que não existam, por padrão ele criará estes arquivos.

Sintaxe:

touch [opções] [arquivos]

Onde:

- arguivos: Arquivos que terão sua data/hora modificados.
- opções: Modificam o comportamento do comando:
 - o -a, --time=atime Faz o touch mudar somente a data e hora do acesso ao arquivo.
 - o -m, --time=mtime Faz o touch mudar somente a data e hora da modificação.
 - o -c, --no-create Não cria arquivos vazios, caso os *arquivos* não existam.
 - o -t MMDDhhmm[AA.ss] Usa Minutos (MM), Dias (DD), Horas (hh), minutos (mm) e opcionalmente o Ano (AA) e

segundos (ss) para modificação dos arquivos, ao invés da data e hora atual.

Exemplos:

```
touch teste Cria o arquivo teste, caso ele não exista.

touch -t 10011230 teste Altera da data e hora do arquivo
para 01/10 e 12:30.

touch -t 120112301999.30 teste Altera da data e hora do arquivo
para 01/12/1999 e 12:30:30.
```

5.120 comando wc

Conta o número de palavras, bytes e linhas em um arquivo ou entrada padrão. Se as opções forem omitidas, o wc mostra a quantidade de linhas, palavras, e bytes.

Sintaxe:

```
wc [opções] [arquivo]
```

Onde:

- arquivo: Arquivo que será verificado pelo comando wc.
- opções: Modificam o comportamento do comando:

```
o -c, --bytes Mostra os bytes do arquivo.
```

- o -w, --words Mostra a quantidade de palavras do arquivo.
- o -1, --lines Mostra a quantidade de linhas do arquivo.

A ordem da listagem das estatísticas é única, e modificando a posição das opções não modifica a ordem que os dados são listados.

Exemplo:

```
wc /etc/passwd Mostra a quantidade de linhas, palavras e letras (bytes) no arquivo /etc/passwd.

wc -w /etc/passwd Mostra a quantidade de palavras.

wc -l /etc/passwd Mostra a quantidade de linhas.

wc -lw /etc/passwd Mostra a quantidade de linhas e palavras.
```

5.130 comando sort

Organiza as linhas de um arquivo texto ou da entrada padrão. A organização é feita por linhas e as linhas são divididas em *campos* que é a ordem que as palavras aparecem na linha separadas por um delimitador (normalmente um espaço).

Sintaxe:

```
sort [opções] [arquivo]
Onde:
```

- Arquivo: É o nome do arquivo que será organizado. Caso não for especificado, será usado o dispositivo de entrada padrão (normalmente o teclado ou um "|").
- Opções:
 - o -b Ignora linhas em branco.
 - o -d Somente usa letras, dígitos e espaços durante a organização.
 - -f Ignora a diferença entre maiúsculas e minúsculas.
 - o -r Inverte o resultado da comparação.
 - -n Caso estiver organizando um campo que contém números, os números serão organizados na ordem aritmética.
 - o −c Verifica se o arquivo já esta organizado. Caso não estiver, retorna a mensagem "disorder on *arquivo*".
 - -i Ignora os caracteres fora da faixa octal ASCII 040-0176 durante a organização.
 - -t caracter Usa caracter como delimitador durante a organização de linhas. Por padrão é usado um espaço em branco como delimitador de caracteres.
 - o +num1 -num2 Especifica qual o campo dentro na linha que será usado na organização. O(s) campo(s) usado(s) para organização estará entre +num1 e +num2. O delimitador padrão utilizado é um espaço em branco (use a opção -t para especificar outro). A contagem é iniciada em "0". Caso não for especificada, a organização é feita no primeiro campo. Caso -num2 não seja especificado, a organização será feita usando a coluna +num1 até o fim da linha.
 - -k num1, num2 Esta é uma alternativa ao método acima para especificar as chaves de organização. O uso é idêntico, mas o delimitador é iniciado em "1".

Exemplos

sort texto.txt - Organiza o arquivo texto.txt em ordem crescente.

sort texto.txt -r - Organiza o conteúdo do arquivo texto.txt em ordem decrescente.

cat texto.txt | sort - Faz a mesma coisa que o primeiro exemplo, só que neste caso a saída do comando cat é redirecionado a entrada padrão do comando sort.

sort -f texto.txt - Ignora diferenças entre letras maiúsculas e minúsculas durante a organização.

sort +1 -3 texto.txt - Organiza o arquivo texto.txt usando como referência a segunda até a quarta palavra (segundo ao quarto campo) que constam naquela linha.

sort -t : +2 -3 passwd - Organiza o arquivo passwd usando como referência a terceira até a quarta palavra (terceiro ao quarto campo). Note que

a opção -t especifica o caracter ":" como delimitador de campos ao invés do espaço. Neste caso, o que estiver após ":" será considerado o próximo campo.

5.140 comando diff

Compara dois arquivos e mostra as diferenças entre eles. O comando diff é usado somente para a comparação de arquivos em formato texto. As diferenças encontradas podem ser redirecionadas para um arquivo que poderá ser usado pelo comando patch para aplicar as alterações em um arquivo que não contém as diferenças. Isto é útil para grandes textos porque é possível copiar somente as modificações (geradas através do diff, que são muito pequenas) e aplicar no arquivo para atualiza-lo (através do patch) ao invés de copiar a nova versão. Este é um sistema de atualização muito usado na atualização dos código fonte do kernel do GNU/Linux.

Sintaxe:

diff [diretório1/arquivo1] [diretório2/arquivo2] [opções]

Onde:

- diretório1/arquivo1 diretório2/arquivo2 Arquivos /diretórios que serão comparados. Normalmente é usado como primeiro arquivo/diretório o mais antigo e o mais novo como segundo.
- Opções:
 - -lines [num] Gera a diferença com [num] linhas de contexto. Por padrão o diff gera um arquivo com 2 linhas que é o mínimo necessário para o correto funcionamento do patch.
 - o -a Compara os dois arquivos como arquivos texto.
 - -b Ignora espaços em branco como diferenças.
 - o -B Ignora linhas em branco inseridas ou apagadas nos arquivos.
 - -I Ignora diferenças entre maiúsculas e minúsculas nos arquivos.
 - o -H Usa análise heurística para verificar os arquivos.
 - N Em uma comparação de diretórios, se o arquivo apenas existe em um diretório, trata-o como presente mas vazio no outro diretório.
 - -P Em uma comparação de diretórios, se o arquivos apenas existe no segundo diretório, trata-o como presente mas vazio no primeiro diretório.
 - -q Mostra somente se os dois arquivos possuem diferenças.
 Não mostra as diferenças entre eles.
 - o -r Compara diretórios e sub-diretórios existentes.
 - -S [nome] Inicia a comparação de diretórios pelo arquivo [nome]. É útil quando cancelamos uma comparação.
 - o -t Aumenta a tabulação das diferenças encontradas.
 - -u Usa o formato de comparação unificado.

Use o comando zdiff para comparar diretamente arquivos compactados pelo utilitário gzip

Use o comando sdiff para visualizar as linhas diferentes entre os dois arquivos em formato texto simples.

Exemplos:

diff texto.txt texto1.txt - Compara o arquivo texto.txt com texto1.txt e exibe suas diferenças na tela.

diff -Bu texto.txt texto1.txt - Compara o arquivo texto.txt com texto1.txt ignorando linhas em branco diferentes entre os dois arquivos e usando o formato unificado.

diff texto.txt texto1.txt >texto.diff - Compara o arquivo texto.txt com texto1.txt e gera um arquivo chamdo texto.diff contendo a diferença entre eles. Este arquivo poderá ser usado pelo patch para aplicar as diferenças existente entre os dois no arquivo texto.txt.

diff -r /usr/src/linux-2.2.13 /usr/src/linux-2.2.14 >patch-2.2.14.diff - Compara o diretório e sub-diretórios linux-2.2.13 e linux-2.2.14 e grava as diferenças entre eles no arquivo patch-2.2.14.diff.

6 PERMISSÕES DE ARQUIVOS

6.1 Conceitos de permissões de arquivos

Como há tipicamente mais de um usuário num sistema Linux, o Linux fornece um mecanismo conhecido com **permissão de arquivos**, que protegem arquivos de usuários de serem mal utilizados por outros usuários. Esse mecanismo permite que arquivos e diretórios "pertençam" ao um usuário particular. Por exemplo, como Larry criou os arquivos em seu diretório home, o Larry é dono desses arquivos e tem acesso a eles.

O Linux também permite que arquivos sejam compartilhados entre usuários e grupos de usuários. Se o Larry desejasse, sele poderia tirar o acesso a seus arquivos de forma a que nenhum outro usuário poderia acessálos. Entretanto, na maioria dos sistemas, o padrão é permitir que outros usuários leiam seus arquivos, mas não possam modificá-los ou exclui-los de alguma forma.

Cada arquivo pertence a um usuário em particular. Entretanto, arquivos também pertencem a um **grupo**, que é um grupo definido de usuários do sistema. Cada usuário é colocado em pelo menos um grupo quando sua conta é criada. No entanto, o administrador do sistema pode conceder ao usuário o acesso a mais de um grupo.

Grupos são usualmente definidos pelo tipo de usuários que acessam a máquina. Por exemplo, numa universidade, o sistema Linux pode ser configurados para os grupos estudante, administracao, faculdade ou convidado. Há também alguns grupos pré-definidos do sistema (como bin e admin), que são usados pelo próprio sistema para controlar acesso aos recursos - muito raramente usuários reais pertencem a grupos de sistema.

Permissões estão divididas em três tipos: leitura, escrita e execução. Essas permissões podem ser concedidas a três tipos de usuários: o dono do arquivo, o grupo ao qual o arquivo pertence, e a todos os usuários, independentemente de grupo.

A permissão de leitura permite que um usuário leia o conteúdo do arquivo e, no caso de diretórios, liste o conteúdo do diretório (usando ls). A permissão de escrita permite que o usuário escreva e/ou modifique o arquivo. Para diretórios, a permissão de escrita permite que o usuário crie novos arquivos ou exclua arquivos naquele diretório. Finalmente, a permissão de execução permite que um usuário rode um arquivo como um programa ou script de shell (o arquivo precisa ser um programa ou script de shell). Para diretórios, ter permissão de execução permite que o usuário entre no diretório em questão (com o comando cd).

6.2 Interpretando permissões de arquivos

Vamos observer um exemplo que demonstra permissões de arquivos. Usando o comando ls com a opção "-l" mostra uma listagem mais completa do arquivo, incluindo permissões de arquivos.

O primeiro campo na lista representa as permissões do arquivo. O terceiro campo é o dono do arquivo (larry) e o quarto campo é o grupo ao qual o arquivo pertence (users). Obviamente, o último campo é o nome do arquivo (stuff). Vamos falar sobre os outros campos mais tarde.

Esse arquivo pertence ao larry, e pertence ao grupo users. A string -rw-r--r-- mostra, em ordem, as permissões concedidas ao dono do arquivo, ao grupo do arquivo, e a todo o resto.

O primeiro caracter da string de permissões ("-") representa o tipo do arquivo. Um hífen ("-") significa que é um arquivo comum (e não um diretório um driver de dispositivo). O próximos três caracteres ("rw-") representam as permissões concedidas ao dono do arquivo, larry. O "r" vem de "read" (leitura) e o "w" de "write" (escrita). Assim, Larry tem permissão de leitura e escrita no arquivo stuff.

Como mencionado, além de permissões de escrita e leitura, há também a permissão de "execução" - representada por um "x". Entretanto, um "-" é listado no exemplo no lugar do "x", indicando que o Larry não tem permissão de execução para este arquivo. Isto está ok, visto que o arquivo stuff não é um programa. Como o Larry é o dono do programa, ele mesmo pode conceder permissão de execução para o arquivo se assim desejar.

Os próximos três caracteres ("r--"), representam as permissões de grupo no arquivo. O grupo que é dono deste arquivo é users. Como somente um "r" aparece aqui, um usuário que pertença ao grupo users pode ler este arquivo.

Os últimos três caracteres (também "r--") representam as permissões concedidas para qualquer usuário do sistema (além do dono do arquivo e daqueles que pertençam ao grupo users). Novamente, como somente um "r" aparece, outros usuários podem ler o arquivo, mas não escrever ou executar.

Aqui vemos outros exemplos de permissões:

-rwxr-xr-x

O dono do arquivo pode ler, escrever e executar o arquivo. Usuários no grupo do arquivo, e todos os outros, podem ler e executar o arquivo

-rw-----

O dono do arquivo pode ler e escrever no arquivo. Nenhum outro usuário pode acessar o arquivo.

-rwxrwxrwx

Todos os usuários podem ler, escrever e executar o arquivo.

6.3 Dependências de Permissões

As permissões concedidas a um arquivo também dependem das permissões do diretório no qual o arquivo está localizado. Por exemplo, mesmo que um arquivo esteja com permissão -rwxrwxrwx, outros usuários não vão poder acessar o arquivo a menos que eles tenham permissão de leitura e

execução no diretório no qual está o arquivo. Por exemplo, se Larry quer restringir acesso a todos os seus arquivos, ele pode configurar as permissões do seu diretório home (/home/larry) para -rwx-----. Desta forma, nenhum outro usuário terá acesso ao seu diretório, e a todos os arquivos e diretórios dentro dele. Larry não precisa se preocupar com as permissões individuais para cada um de seus arquivos.

Em outras palavras, para acessar um arquivo, você tem que ter permissão de execução para todos os diretórios no caminho, e permissão de leitura (ou execução) para o arquivo.

Tipicamente, usuários num sistema Linux são bastante descuidads com seus arquivos. A permissão usual dada a arquivos é -rw-r--r-, que permite que outros usuários leiam os arquivos mas não modifiquem. A permissão usual dada a diretórios é -rwxr-xr-x, que permite que outros usuários entrem em seus diretórios, mas não criem ou apaguem arquivos dentro deles.

Entretanto, muitos usuários desejam manter outros usuários longe de seus arquivos. Colocando as permissões de um arquivo para -rw----- irá previnir que qualquer outro usuário acesse o arquivo. Da mesma forma, se um diretório tiver a permissão -rwx-----, nenhum outro usuário poderá entrar nele.

6.4 Modificando permissões

O comando "chmod" é usado para modificar permissões de um arquivo. Somente o dono do arquivo pode mudar as permissões para o arquivo. A sintaxe do chmod é "chmod {a,u,g,o}{+,-}{r,w,x} arquivos"

Brevemente, você fornece uma ou mais letras indicando para quem você está concedendo ou retirando permissões: todos (a, do inglês, all); usuário (u), grupo (g), ou outros (o). Depois você especifica se está adicionando (+) ou removendo (-) direitos. E finalmente você especifica o tipo de permissão: leitura (read), escrita (write), ou execução. Alguns exemplos de comandos possíveis são:

chmod a+r stuff

Dá permissão de leitura para todos os usuários.

chmod +r stuff

A mesma coisa - se nenhuma ${\bf a}, \ {\bf u}, \ {\bf g}$ ou ${\bf o}$ é especificado, ${\bf a}$ é assumido.

chmod og-x stuff

Remove permissão de execução de usuários que não o próprio dono.

chmod u+rwx stuff

Permite que o dono leia, escreva e execute o arquivo.

chmod o-rwx stuff

Remove as permissões de leitura, escrita e execução para outros usuários que não sejam o dono ou usuários do grupo.

7 GERENCIANDO LINKS DE ARQUIVOS

Links permitem que você dê mais de um nome a um único arquivo. Arquivos são identificados pelo sistema pelo número do "inode", que é o identificador único para o arquivo no sistema. Um diretório é na verdade uma listagem de "inodes" com seus nomes de arquivos correspondentes. Cada nome de arquivo num diretório é um **link** para um "inode" particular.

7.1 Links Rígidos

O comando "In" é usado para criar múltiplos links para um arquivo. Por exemplo, digamos que você tenha um arquivo chamado "foo" em um diretório. Usando "Is -i", você pode ver o número do "inode" para este arquivo.

```
/home/larry# ls -i foo
22192 foo
/home/larry#
```

Aqui, "foo" tem o número de inode 22192 no sistema de arquivos. Você pode criar outro link para o arquivo "foo", chamado, por exemplo, "bar", como seque:

```
/home/larry# ln foo bar
```

Com o "ls -i", você pode ver que os dois arquivos tem o mesmo inode.

```
/home/larry# ls -i foo
22192 bar 22192 foo
/home/larry#
```

Agora, especificando tanto "foo" quanto "bar", você terá acesso ao mesmo arquivo. Se você fizer mudanças a "foo", essas mudanças aparecem em "bar" também. Para todos os proósitos, "foo" e "bar" são o mesmo arquivo.

Esses links são conhecidos como **links rígidos** (**hard link**), porque criam um link direto para um inode. Note que você só pode fazer hard links para arquivos se os links estiverem no mesmo sistema de arquivos. Links simbólicos (veja abaixo) não têm esta restrição.

Quando você exclui um arquivo com "rm", você está, na verdade, apagando um link para o arquivo. Se você usar o comando

```
/home/larry# rm foo
```

somente o link chamado foo é excluído, bar ainda vai existir. Um arquivo é realmente excluído do sistema somente quando não há mais links para ele. Usualmente, arquivos só tem um link. Assim, usando o comando "rm" exclui o arquivo. Entretanto, se um arquivo tem múltiplos links, usando o comando "rm" excluirá somente um link. Para excluir o arquivo, você tem que excluir todos os links para o mesmo.

O comando "ls -l" mostra o número de links para um arquivo (entre outras informações).

```
/home/larry# ls -l foo bar
-rw-r--r- 2 root root 12 Aug 5 16:51 bar
```

```
-rw-r--r-- 2 root root 12 Aug 5 16:50 foo /home/larry#
```

A segunda coluna na listagem, "2", especifica o número de links para o arquivo.

Com isso claro, um diretório é, na verdade, somente um arquivo contendo a informação sobre as associações link para inode. Além disso, cada diretório contém, no mínimo, dois links rígidos: "." (um link apontando para si mesmo), e ".." (um link apontando para o diretório superior). No diretório root (/), o link ".." aponta novamente para /. (Em outras palavras, o diretório superior do diretório root é o próprio diretório root).

7.2 Links Simbólicos

Links simbólicos, ou **symlinks**, são outro tipo de link, e são diferentes dos links rígidos. Um link simbólico permite que você dê outro nome a um arquivo, mas não faz o link ao arquivo por inode.

O comando "In -s" cria um link simbólico para um arquivo. Por exemplo, se você usa o comando:

```
/home/larry# ln -s foo bar
```

Você estará criando um link simbólico chamado "bar" que aponta para o arquivo "foo". Se você usa o "ls -i", você verá que os dois arquivos têm, de fato, inodes diferentes.

```
/home/larry# ls -i foo bar
22195 bar 22192 foo
/home/bar#
```

Entretanto, usando o "ls -l", vemos que o arquivo "bar" é um ponteiro simbólico para "foo".

```
/home/larry# ls -l foo bar
lrwxrwxrwx 1 root root 3 Aug 5 16:51 bar -> foo
-rw-r--r-- 1 root root 12 Aug 5 16:50 foo
/home/larry#
```

As permissões de arquivo num link simbólico não são usadas (sempre aparecem como rwxrwxrwx). Elas são determinadas pelas permissões no arquivo alvo do link simbólico (em nosso exemplo, o arquivo foo).

Funcionalmente, links rígidos e links simbólicos são similares, mas há diferenças. Você pode, por exemplo, criar um link simbólico para um arquivo que não existe; o mesmo não funciona para links rígidos. Links simbólicos são processados pelo kernel diferentemente de links rígidos. É somente uma diferença técnica, mas algumas vezes importante. Links simbólicos são úteis porque identificam o arquivo para o qual apontam. Com links rígidos, não há uma maneira fácil de determinar quais arquivos estão "linkados" para o mesmo inode.

Links são usados em muitos lugares no sistema Linux. Links simbólicos são especialmente importante para bibliotecas compartilhadas no /lib.

8 COMANDOS DE BUSCA

8.1 O comando grep

Procura por um texto dentro de um arquivo(s) ou no dispositivo de entrada padrão.

Sintaxe:

grep [expressão] [arquivo] [opções]

Onde:

- Expressão: Expressão regular que será procurada no texto. Se tiver mais de 2 palavras você deve identifica-la com aspas "" caso contrário o grep assumirá que a segunda palavra é o arquivo!
- Arquivo: Arquivo onde será feita a procura.
- Opções:
 - o -A [número] Mostra o [número] de linhas após a linha encontrada pelo grep.
 - o -B [número] Mostra o [número] de linhas antes da linha encontrada pelo grep.
 - o -f [arquivo] Especifica que o texto que será localizado, esta no arquivo [arquivo].
 - o -h, --no-filenameNão mostra os nomes dos arquivos durante a procura.
 - o -i, --ignore-caselgnora diferença entre maiúsculas e minúsculas no texto procurado e arquivo.
 - o -n, --line-number Mostra o nome de cada linha encontrada pelo grep.
 - o -U, --binary Trata o arquivo que será procurado como binário.

Se não for especificado o nome de um arquivo ou se for usado um hífen "-", grep procurará a string no dispositivo de entrada padrão. O grep faz sua pesquisa em arquivos texto. Use o comando zgrep para pesquisar diretamente em arquivos compactados com gzip, os comandos e opções são as mesmas.

Exemplos:

grep "capitulo" texto.txt ps ax | grep inetd grep "capitulo" texto.txt -A 2 -B 2

8.2 O comando find

Procura por arquivos/diretórios no disco. find pode procurar arquivos através de sua data de modificação, tamanho, etc através do uso de opções. find, ao contrário de outros programas, usa opções longas através de um "-". Sintaxe:

milaxo.

find [diretório] [opções/expresão]

Onde:

- Diretório: Inicia a procura neste diretório, percorrendo seu subdiretórios.
- opções/expressão:
 - -name [expressão] Procura pelo nome [expressão] nos nomes de arquivos e diretórios processados.
 - depth Processa os sub-diretórios primeiro antes de processar os arquivos do diretório principal.
 - -maxdepth [num] Faz a procura até [num] sub-diretórios dentro do diretório que está sendo pesquisado.
 - -mindepth [num] Não faz nenhuma procura em diretórios menores que [num] níveis.
 - -mount, -xdev Não faz a pesquisa em sistemas de arquivos diferentes daquele de onde o comando find foi executado.
 - -amin [num] Procura por arquivos que foram acessados [num] minutos atrás. Caso for antecedido por "-", procura por arquivos que foram acessados entre [num] minutos atrás até agora.
 - -atime [num] Procura por arquivos que foram acessados [num] dias atrás. Caso for antecedido por "-", procura por arquivos que foram acessados entre [num] dias atrás e a data atual.
 - -gid [num] Procura por arquivos que possuam a identificação numérica do grupo igual a [num].
 - -group [nome] Procura por arquivos que possuam a identificação de nome do grupo igual a [nome].
 - -uid [num] Procura por arquivos que possuam a identificação numérica do usuário igual a [num].
 - -user [nome] Procura por arquivos que possuam a identificação de nome do usuário igual a [nome].
 - -inum [num] Procura por arquivos que estão localizados no inodo [num].
 - -links [num] Procura por arquivos que possuem [num] links como referência.
 - -mmin [num] Procura por arquivos que tiveram seu conteúdo modificado há [num] minutos. Caso for antecedido por "-", procura por arquivos que tiveram seu conteúdo modificado entre [num] minutos atrás até agora.
 - -mtime [num] Procura por arquivos que tiveram seu conteúdo modificado há [num] dias. Caso for antecedido por "-", procura por arquivos que tiveram seu conteúdo modificado entre [num] dias atrás até agora.
 - -nouser Procura por arquivos que não correspondam a identificação do usuário atual.

- -nogroup Procura por arquivos que não correspondam a identificação do grupo do usuário atual.
- -perm [modo] Procura por arquivos que possuam os modos de permissão [modo]. Os [modo] de permissão pode ser numérico (octal) ou literal.
- -used [num] O arquivo foi acessado [num] vezes antes de ter seu status modificado.
- -size [num] Procura por arquivos que tiverem o tamanho [num]. [num] pode ser antecedido de "+" ou "-" para especificar um arquivo maior ou menor que [num]. A opção size pode ser seguida de:
 - § b Especifica o tamanho em blocos de 512 bytes. É o padrão caso [num] não seja acompanhado de nenhuma letra.
 - § c Especifica o tamanho em bytes.
 - § k Especifica o tamanho em Kbytes.
- -type [tipo] Procura por arquivos do [tipo] especificado.
 Os seguintes tipos são aceitos:
 - § b bloco
 - § c caracter
 - § d diretório
 - § p-pipe
 - § f arquivo regular
 - § I link simbólico
 - § s sockete

A maior parte dos argumentos numéricos podem ser precedidos por "+" ou "-". Para detalhes sobre outras opções e argumentos, consulte a página de manual.

Exemplos:

find / -name grep - Procura no diretório raíz e sub-diretórios um arquivo/diretório chamado grep.

 $\label{find preposition} \mbox{find / -name grep -maxdepth 3 - Procura no diretório raíz e subdiretórios até o 3o. nível, um arquivo/diretório chamado grep.}$

find . -size +1000k - Procura no diretório atual e sub-diretórios um arquivo com tamanho maior que 1000 kbytes (1Mbyte).

find / -mmin 10 - Procura no diretório raíz e sub-diretórios um arquivo que foi modificado há 10 minutos atrás.

find / -links 4 - Procura no diretório raíz e sub-diretórios, todos os arquivos que possuem 4 links como referência.

8.3 O comando which

Mostra a localização de um arquivo executável no sistema. A pesquisa de arquivos executáveis é feita através do path do sistema.

Sintaxe:

which [comando]
Exemplos:
which ls
which shutdown
which which

8.4 O comando whereis

Localiza o arquivo que contém uma página de manual. A pesquisa é feita usando-se os caminhos de páginas de manuais configuradas no sistema (normalmente o arquivo /etc/manpath.config).

Sintaxe:

whereis [comando]

Exemplos:

whereis ls whereis cd

9 USANDO O EDITOR VI

Um editor de textos é um programa usado para editar arquivos que são compostos de textos: uma carta, um programa C, ou um arquivo de configuração do sistema. Ainda que existam vários desses editores disponíveis para o Linux, o único que você garantidamente vai encontrar em qualquer UNIX ou sistema Linux, é o vi (do inglês, "visual editor"). O vi não é o editor mais fácil de usar, nem é muito auto-explicativo. Entretanto, como o vi é tão comum em sistemas UNIX/Linux, e muitas vezes necessário, ele merece um pouco de nossa atenção.

Sua escolha por um editor é mais uma questá pessoal de gosto e estilo. Muitos usuários preferem o barroco, auto-explicativo e poderoso Emacs - um editor com mais funções que qualquer outro programa no mundo UNIX. Por exemplo, o Emacs tem o seu próprio dialeto da linguagem de programação LISP embutido, e tem muitas outras extensões (uma das quais é um tipo Eliza - um programa de inteligência artificial). No entanto, como o Emacs e seus arquivos de suporte são relativamente grandes, ele não pode ser instalado em alguns sistemas. O vi, por outro lado, é pequeno e poderoso, mas mais difícil de usar. Entretanto, uma vez que você saiba lidar com ele, será muito fácil.

Esta seção apresenta uma introdução ao vi - não vamos discutir todas os seus atributos, somente aqueles que você precisa saber para começar a usá-lo. Você pode verificar a página de manual do vi se estiver interessado em aprendar mais sobre o editor e suas funções. Alternativamente, você pode ler o livro Learning the vi Editor, da editora O'Reilly and Associates, ou Vi Tutorial, da Specialized Systems Consultants (SSC) Inc.

9.1 Conceitos.

Enquanto estiver usando o vi, em qualquer momento você está em um dos três modos de operação. Esses modos são chamados: *modo de comando*, *modo de inserção*, e *modo última linha*.

Quando você inicia o vi, você está no modo de comando. Esse mode permite que você use comandos para editar arquivos ou para trocar para outros modos. Por exemplo, digitando "x" no modo comando apaga o caracter que está sob o cursor. As teclas de setas movem o cursor pelo arquivo que você está editando. Geralmente, os comandos usados no modo de comando são de um ou dois caracteres.

Você pode inserir ou editar textos no modo de inserção. Quando estiver usando o vi, vai provavelmente usar a maior parte do tempo neste modo. Você inicia o modo de inserção pressionando o comando "i" (de "inserir") no modo de comando. Enquanto estiver em modo de inserção, você pode inserir texto no documento na posição do cursor. Para sair do modo de inserção e retornar ao modo de comando, pressione <Esc>.

O modo de última linha é um modo especial que permite alguns comandos extendidos do vi. Ao digitar esses comandos, eles aparecem na última linha da tela (daí o nome). Por exemplo, quando você digita ":" no modo de comando, você pula para o modo de última linha e pode usar os comandos "wq" (gravar e sair - do inglês, write and quit), ou "q!" (para sair sem salvar). O

modo de última linha é geralmente usado para comandos do vi que são mais longos do que um caracter. No modo de última linha, você entra um comando de uma linha e pressiona <Enter> para executá-lo.

9.2 Iniciando o vi

A melhor maneira de compreender estes conceitos é chamar o vi e editar um arquivo. O exemplo "screens" abaixo mostra algumas linhas de texto, como se a tela tivesse somente seis linhas de altura, ao invés de vinte e quatro.

A sintaxe do vi é

vi arquivo

onde arquivo é o nome do arquivo a editar.

Chame o vi digitando:

/home/larry# vi test

para editar o arquivo "test". Você deve ver algo parecido com:

```
~
~
~
~
~
~
~
~
~
"test" [New file]
```

A coluna de caracteres "~" indicam que você está no fim do arquivo. No exemplo, o caracter "_" representa o cursor.

9.3 Inserindo texto

O programa vi está agora no modo de comando. Insira um texto no arquivo - pressionando "i" você coloca o editor em modo de inserção. A partir daí, comece a digitar.

```
Now is the time for all good men to come to the aid of the party
~
~
~
~
~
~
~
~
```

```
~ ~
```

Digite quantas linhas você quiser (pressionando <Enter> depois de cada uma). Você pode corrigir erros com a tecla <Back Space>.

Para terminar o modo de inserção e retornar ao modo de comando, pressione <Esc>.

No modo de comando você pode usar as setas para mover pelo arquivo. (Se você tem somente uma linha de texto, e tentar usar as teclas para cima e para baixo, vai sentir que vi vai tocar um bip).

Há muitas maneiras de inserir um texto que não o comando "i". O comando "a" insere um texto começando após a posição atual do cursor, ao invés de na posição atual do cursor. Por exemplo, use a seta da esquerda para mover o cursor entre as palavras "good" e "men".

Pressione "a" para entrar no modo de inserção, digite "wo", e então pressione <Esc> para retornar ao modo comando.

Para começar a inserir texto na próxima linha, use o comando "o". Pressione "o" e entre uma ou duas linhas mais:

```
Now is the time for all good humans to come to the aid of the party.

Afterwards, we'll go out for pizza and beer.
```

9.4 Apagando texto

A partir do modo de comando, o comando "x" apaga o caracter sob o cursor. Se você pressionar cinco vezes "x", vai chegar em:

Agora pressione "a" e insira algum texto, seguido de <esc>:

Você pode apagar linhas inteiras usando o comando "dd" (isso mesmo, pressione "d" duas vezes seguidas), Se o cursor está na segunda linha, e você pressionar "dd", verá:

Para apagar a palavra onde está o cursor, use o comando "dw". Coloque o cursor no início da palavra good, e digite "dw".

9.5 Modificando texto

Você pode substituir seções de texto usando o comando "R" (maiúsculo). Posicione o cursor na primeira letra da palavra "party", pressione "R" e digite a palavra "hungry".

Usando "R" para editar um texto é como usar "i" ou "a", mas o "R" sobreescreve, ao invés de inserir.

O comando "r" (minúsculo) substitui o caracter sob o cursor. Por exemplo, mova o cursor para o início da palavra "Now" e pressione "r", depois "C", você verá:

O comando "~" troca a caixa da letra que está sob o cursor de alta para baixa e vice-versa. Por exemplo, se você posiciona o cursor no "o", na palvra "Cow" acima e repetidamente pressiona "~", terá:

Nota: em teclados programados para o português, você terá que pressionar "~" seguido da tecla de espaço para ter o mesmo efeito.

9.6 Comandos para movimentar o cursor

Você já sabe como usar as setas para se movimentar pelo documento. Além disso, você pode usar os comandos "h", "j", "k" e "l" para mover o cursor para a esquerda, baixo, cima e direita, respectivamente. Isso é útil quando (por alguma razão) suas teclas de setas não funcionam corretamente.

O comando "w" move o cursor para o início da próxima palavra, o comando "b" para o início da palavra anterior.

O comando "0" (tecla zero) move o cursor para o início da linha corrente, e o comando "\$" para o fim da linha.

Quando editar arquivos grandes, você vai querer avançar e retroceder através do arquivo uma tela por vez. Pressionado "Ctrl-F" move o cursor uma tela para a frente, e Ctrl-B, uma tela para trás.

Para mover o cursor para o fim do arquivo, pressione "G". Você também pode movê-lo para uma linha arbitrária; por exemplo, digite o comando "10G" para mover o cursor para a linha 10 no arquivo. Para movê-lo para a primeira linha, use "1G".

Você pode combinar comandos de movimentação com outros comandos, como aqueles usados para apagar texto. Por exemplo, o comando "d\$" apaga tudo do cursor até o fim da linha; "dG" apaga tudo do cursor até o fum do arquivo, e assim por diante.

9.7 Salvando arquivos e saindo do vi

Para sair do vi sem salvar mudanças no arquivo, use o comando ":q!". Quando você pressiona ":", o cursor se move para a última linha da tela e você estará no modo de última linha.

No modo de última linha, certos comandos extendidos estão disponíveis. Um deles é o "q!", que sai do vi sem salvar. O comando ":wq" salva o arquivo e sai do vi. O comando "ZZ" (do modo comando, sem os ":") é equivalente a ":wq". Se o arquivo não foi modificado desde a última vez que foi salvo, o comando só sai, preservando a data de modificação da última mudança. Lembre-se que você deve pressionar <Enter> depois de um comando entrado no modo de última linha.

Para salvar o arquivo sem sair do vi, use ":w".

9.8 Editando outro arquivo

Para editar outro arquivo, use o comando ":e". Por exemplo, para parar a edição do arquivo "test" e editar o arquivo "foo" no lugar, use o comando:

Se você usa ":e" sem salvar o primeiro arquivo, você vai receber a mensagem de erro:

```
No write since last change (":edit!" overrides)
```

que significa que o vi não quer editar outro arquivo até que você salve o primeiro. Neste ponto, você pode usar ":w" para salvar o arquivo original, e depois o ":e", ou pode usar o comando:

O sinal "!" diz ao vi que você realmente quer fazer isso - editar o novo arquivo sem salvar as mudanças do primeiro.

9.9 Incluindo outros arquivos

Se você usar o comando ":r", pode incluir o conteúdo de outro arquivo no arquivo corrente. Por exemplo, o comando

```
:r foo.txt
```

insere o conteúdo do arquivo "foo.txt" no texto atul na localização do cursor.

9.10 Rodando comandos do shell

Você também pode rodar comandos do shell de dentro do vi. O comando ":r!" funciona como ":r", mas ao invés de ler um arquivo, ele insere a saída do comando dado na posição corrente do cursor. Por exemplo, se você usar o comando:

```
r! ls -F
```

você vai terminar com:

```
Cow is the time for all humans to come to the aid of the hungry.

letters/
misc/
papers/
~
~
```

Você também pode rodar um comando do shell de dentro do vi sem que o texto seja inserido no arquivo em edição. Por exemplo, se usar o comando :! ls -F

O "Is -F" será executado e o resultado mostrado na tela, mas não inserido no arquivo que você está editando. Se você usar o comando :shell

O vi inicia uma instância do shell, permitindo que você ponha o vi temporariamente em segundo plano, enquanto executa outros comandos. Simplesmente dê um logout (usando o comando exit) do shell para retornar ao vi.

9.11 Ajuda no vi

O vi não fornece muita ajuda interativa (a maioria dos programas Linux também não), mas você sempre pode ler a página do manual para o vi. O vi é uma fachada para o editor "ex", que manipula muitos dos comandos do modo de última linha do vi. Assim, além de ler a página de manual para o vi, veja também para o "ex".

Você também pode usar o comando ":help" para ver uma ajuda dos comandos básicos na tela.

10 ACESSANDO ARQUIVOS MS-DOS

Se, por qualquer motivo bizarro, você quiser acessar arquivos do MS-DOS (ou Windows), pode fazê-lo facilmente no Linux.

A maneira usual de acessar arquivos MS-DOS é montar uma partição ou disquete MS-DOS no Linux, permitindo que você acesse os arquivos diretamente através do sistema de arquivos. Por exemplo, se você tem um disquete MS-DOS em /dev/fd0, o comando

mount -t msdos /dev/fd0 /mnt

vai montá-lo no diretório /mnt.

Você também pode montar uma partição MS-DOS no seu disco rígido para acessá-la através do Linux. Se você tem uma partição MS-DOS em /dev/hda1, o comando:

mount -t msdos /dev/hda1 /mnt

monta a partição. Lembre-se de desmontar a partição quando tiver terminado de usá-la. Você pode ter uma partição MS-DOS montada automaticamente na inicialização do sistema se você incluir a entrada no /etc/fstab. A próxima linha no /etc/fstab irá montar a partição MS-DOS no /dev/hda1 no diretório /dos automaticamente:

/dev/hda1 /dos msdos defaults

Você também pode montar o sistema de arquivos VFAT (usado pelo Windows 95):

mount -t vfat /dev/hda1 /mnt

Isso permite acessar nomes de arquivo longos do Windows. Você pode montar um sistema de arquivos FAT16 e usar isto para gerar nomes de arquivo longos.

O software Mtools também pode ser usado para acessar arquivos MS-DOS. Os comandos mcd, mdir e mcopy se comportam como seus correspondentes MS-DOS. Se você instalar o Mtools, poderá consultar páginas de manual para esses comandos.

Acessar arquivos MS-DOS é uma coisa, executá-los é outra. Há um emulador MS-DOS em desenvolvimento no Linux. Ele é incluído em muitas distribuições. Também pode ser baixado de muitos sites de FTP. O emulador é bastante poderoso e pode rodar várias aplicações. Entretanto, Linux e MS-DOS são sistemas bem diferentes. O poder de qualquer emulador MS-DOS num UNIX é limitado. Há também um emulador Windows para ambiente X em desenvolvimento.

11 O SHELL

11.1 Tipos de shell

Como mencionado anteriormente, o Linux é um sistema de arquivos multi-tarefa e multi-usuário. Ser multi-tarefa é bem útil, e uma vez que você tenha entendido isso, vai usá-lo todo o tempo. Em pouco tempo você vai estar rodando programas em segundo plano, mudando entre tarefas, e fazendo pipelines entre programas. Tudo isso junto para atingir resultados complicados em um único comando.

Muitas das características que nós vamos cobrir nesta seção são fornecidas pelo próprio shell. Tome cuidado para não confundir o Linux (o próprio sistema operacional) com um shell - um shell é somente uma interface para o sistema que está embaixo. O shell fornece funcionalidade para o Linux.

Um shell não é somente um interpretador para comandos interativos que você digita no prompt, mas também uma poderosa linguagem de programação. Ele permite que você escreva scripts de shell, para agrupar vários comandos shell em um único arquivo. Se você sabe MS-DOS, você vai reconhecer a semelhança com os arquivos "batch" (.BAT). Os scripts do shell são uma ferramenta poderosa, que permitirão que você automatize e expanda seu uso do Linux.

Há vários tipos de shell no mundo Linux. Os dois tipos principais são o "Bourne shell" e o "C shell". O "Bourne shell" usa uma sintaxe de comando como o shell original dos sistemas UNIX, como o System III. O nome do "Bourne shell" na maioria dos sistemas Linux é /bin/sh (onde sh vem de "shell"). O "C shell" usa uma sintaxe diferente, parecida com programação C, e na maioria dos sitemas é chamado de /bin/csh.

No Linux, várias variantes desses shell estão disponíveis. As duas mais comunmente usadas são "Bourne Again Shell", ou "Bash" (/bin/bash), e "Tcsh" (/bin/tcsh). Bash é uma forma de Bourne shell que inclui muitas das características avançadas encontradas no "C shell". Como o bash suporta um conjunto maior de comandos que o "Bourne shell", os scripts em shell escritos para o Bourne shell devem funcionar no bash. Se você prefere usar a sintaxe do C shell, o Linux suporta o tcsh, que é uma versão expandida do C shell original.

O tipo de shell que você vai escolher é mais uma questão de religião. Alguns preferem a sintaxe do Bourne shell com as características avançadas do bash, e outros preferem a sintaxe mais estruturada do C shell. Para comandos normais como cp e ls, o shell que você usar não faz diferença - a sintaxe não muda. Só quando você começar a escrever scripts shell ou usar funções avançadas de um shell é que as diferenças entre os tipos vai começar a importar.

A medida que for se acostumando, você vai notar diferenças entre o Bourne e o C shell. Entretanto, para os propósitos deste livro, a maioria das diferenças é irrelevante. (Se você está curioso sobre este ponto, leia as páginas de manual do bash e do tcsh).

11.2 Caracteres Coringa

Uma característica fundamental da maioria dos shell do Linux é a habilidade de se referir a mais de uma arquivo usando caracteres especiais. Esses **caracteres coringa** permitem que você se refira, digamos, a todos os arquivos que contenham o caracter "n".

O caracter coringa "*" especifica qualquer caracter ou seqüência de caracteres num nome de arquivo. Quando você usa o caracter "*" em um nome de arquivo, o shell o substitui por todas as possibilidades possíveis de nomes de arquivos no diretório ao qual você se refere.

Aqui temos um exemplo rápido. Suponha que Larry tenha os arquivos frog, joe e stuff no seu diretório corrente.

```
/home/larry# Is
frog joe stuff
/home/larry#
```

Para especificar todos os arquivos contendo a letra "o" no nome, use o comando:

```
/home/larry# Is *o*
frog joe
/home/larry#
```

Como você pode ver, cada instância do "*" é substituída por todas as substituições que casam com os arquivos no diretório corrente.

O uso do "*" simplesmente, casa com todos os nomes, pois todos os caracteres casam com o caracter coringa.

```
/home/larry# Is * frog joe stuff /home/larry#
```

Abaixo, mais alguns exemplos:

```
/home/larry# 1s f*
frog
/home/larry# 1s *ff
stuff
/home/larry# 1s *f*
frog stuff
/home/larry# 1s s*f
stuff
/home/larry# 1s ry#
```

O processo de trocar um "*" numa série de nomes de arquivos é chamado de **expansão de caracteres coringa** e é feito pelo shell. Isto é importante: um comando individual, como ls, *nunca* vê o "*" na sua lista de

parâmetros. O shell expande os caracteres coringa para incluir os nomes de arquivos que casam. Assim, o comando:

```
/home/larry# 1s *o*
```

```
é expandido pelo shell para:
```

```
/home/larry# 1s frog joe
```

Uma nota importante sobre o caracter coringa "*" é que ele não expande para nomes que começam com um ponto ("."). Esses arquivos são tratados como ocultos - apesar de não serem realmente ocultos, eles não são mostrados numa listagem ls normal nem são tocados pelo uso do caracter "*".

Aqui temos um exemplo. Nós mencionamos antes que cada diretório contém duas entradas especiais: ".", que se refere ao diretório corrente, e "..", que se refere ao diretório superior. Entretanto, quando você usao o ls, essas entradas não são mostradas.

```
/home/larry# 1s
frog joe stuff
/home/larry#
```

Se você usa a opçao -a no ls, entretanto, você pode ver nomes de arquivos que começam com ".". Observe:

```
/home/larry# 1s-a . .. .bash_profile .bashrc frog joe stuff /home/larry#
```

A lista contém as duas entradas especiais, "." e "..", assim como outros arquivos ocultos - .bash_profile e .bashrc. Esses dois arquivos são arquivos de inicialização usados pelo bash quando o Larry entra no sistema.

Note que quando você usa o caracter coringa "*", nenhum dos arquivos começando com "." são mostrados.

```
/home/larry# 1s *
frog joe stuff
/home/larry#
```

Essa é uma característica de segurança: se o "*" encontrasse nomes começando com ".", ele também encontraria os diretórios "." e "..". Isso pode ser perigoso quando ao usar alguns comandos.

Outro caracter coringa é o "?". O "?" expande para um único caracter. Assim, "ls ?" mostra todos os arquivos com nomes de um caracter só. E "ls termca?" mostraria "termcap", mas não "termcap.backup". Segue outro exemplo:

```
/home/larry# 1s j?e joe
```

```
/home/larry# 1s f??g
frog
/home/larry# 1s ????f
stuff
/home/larry#
```

Como pode ver, caracteres coringa permitem que você especifique muitos arquivos de uma só vez. Já comentamos que os comandos cp e mv podem copiar ou mover mais de um arquivo por vez. Por exemplo:

```
/home/larry# cp/etc/s*/home/larry
```

copia todos os arquivos começando com "s" para o diretório "/home/larry". O formato do comado cp é

```
cp {arq1} {arq2} {arq3} ... {arqN} {destino}
```

onde {arq1} até {arqN} é uma lista de nomes de arquivos a copiar, e destino é o arquivo ou diretório destino. "mv" tem uma sintaxe idêntica.

Se você está copiando ou movendo mais de um arquivo, o destino deve ser um diretório. Você somente pode copiar ou mover um único arquivo para outro arquivo.

11.3 Redirecionamentos e pipes

11.3.1 Entrada padrão e saída padrão

Muitos comandos no Linux recebem dados de entrada pelo que chamamos entrada padrão (standard input) e dão saída na saída padrão (standard output) - freqüentemente abreviadas para stdin e stdout. O shell trabalha de forma a que sua stdin seja o teclado e sua stdout seja a tela.

Segue um exemplo usando o comando cat. Normalmente, o cat lê dados de todos os arquivos especificados na linha de comando e envia dados diretamente para a stdout (normalmente sua tela). Assim, o comando

/home/larry/papers# cat history-final masters-thesis

mostra o conteúdo do arquivo history-final seguido pelo conteúdo do arquivo masters-thesis

No entanto, se você não especifica um nome de arquivo, o cat lê dados da stdin e manda dados para a stdout. Veja o exemplo:

```
/home/larry/papers# cat
Hello there.
Hello there.
Bye.
Bye.
{Ctrl-D}
```

/home/larry/papers#

Cada linha que você digita é imediatamente ecoada pelo cat. Quando estiver lendo da entrada padrão, você indica que a entrada "terminou" mandando um sinal de EOT (End-Of-Text - fim de texto) que, em geral, é gerado ao pressionar Ctrl-D.

Aqui você tem outro exemplo. O comando sort lê linhas de texto (novamente, da stdin, a menos que você especifique um ou mais nomes de arquivos) e manda-as ordenadas para a stdout. Tente o seguinte:

```
/home/larry/papers# sort
bananas
carrots
apples
{Ctrl-D}
apples
bananas
carrots
/home/larry/papers#
```

Agora você pode ordenar sua lista de compras.... Linux é útil ou não é?

11.3.2 Redirecionando entrada e saída

Digamos que você queira mandar a saída do sort para um arquivo, para gravar sua lista de compras no disco. O shell permite que você redirecione a saída padrão para um arquivo usando o símbolo ">". Veja como funciona:

```
/home/larry/papers# sort > shopping-list
bananas
carrots
apples
{Ctrl-D}
/home/larry/papers#
```

Como você pode notar, o resultado do comando sort não é mostrado, mas sim gravado num arquivo chamado shopping-list. Vamos dar uma olhada no arquivo:

```
/home/larry/papers# cat shopping-list apples bananas carrots /home/larry/papers#
```

Agora você pode ordenar sua lista de compras e salvá-la também! Mas vamos supor que você esteja armazenando a lista original não ordenada no

arquivo itens. Uma maneira de ordenar a informação e gravá-la num arquivo seria dar ao comando sort o nome do arquivo a ser lido no lugar da stdin, e redirecionar a stdout assim como fizemos antes. Veja o exemplo:

/home/larry/papers# sort itens > shopping-list /home/larry/papers# cat shopping-list apples bananas carrots /home/larry/papers#

Entretanto, há outra forma de fazer isto. Além de redirecionar a stdout, você pode redirecionar a stdin também, usando o símbolo "<".

/home/larry/papers# sort < itens apples bananas carrots /home/larry/papers#

Tecnicamente, "sort < itens" é equivalente a "sort itens", mas permite demonstrar o seguinte ponto: "sort < itens" se comporta como se os dados no arquivo "itens" tivessem sido digitados na stdin. O shell manipula o redirecionamento. O "sort" não recebeu o nome do arquivo itens para ler; tudo o que o sort sabe é que recebeu uma lista da stdin, como se você tivesse digitado todos os dados pelo teclado.

Isso introduz o conceito de um **filtro**. Um filtro é um programa que lê dados da stdin, processa-os de alguma forma, e manda-os processados para a stdout. Usando redirecionamentos, stdin e stdout podem ser referenciados de arquivos. Como mencionado acima, stdin e stdout, por padrão, se comportam como teclado e tela respectivamente. O "sort" é um filtro simples. Ele ordena os dados de entrada e manda o resultdao para a saída padrão. O "cat" é ainda mais simples. Ele não faz nada com os dados de entrada. Simplesmente mostra o que lê.

11.3.3 Redirecionamento de saída não -destrutivo

Usando ">" para redirecionar a saída para um arquivo é destrutivo. Em outras palavras, o comando

/home/larry/papers# ls > arquivo-lista

sobreescreve o conteúdo do arquivo arquivo-lista. Se, no lugar de usar ">", você usar o símbolo ">>", a saída é adicionada ao final do arquivo dado ao invés de sobreescrevê-lo. Por exemplo,

/home/larry/papers# ls >> arquivo-lista

anexa a saída do comando ls ao arquivo arquivo-lista.

Mantenha em mente que redirecionamentos e pipes são características do shell - que suporta o uso de ">", ">>" e "|". Não tem nada a ver com os comandos mesmos.

11.3.4 Usando pipes

Já demonstramos como usar o "sort" como um filtro. Entretanto, esses exemplos assumem que você tenha dados armazenados em algum arquivo ou que vá digitar os dados da entrada padrão você mesmo. E se os dados que você quer ordenar vierem da saída de outro comando, por exemplo o ls?

A opção -r do "sort" ordena os dados em ordem alfabética reversa. Se você quer listar os arquivos no diretório corrente em ordem reversa, uma maneira de fazer isso seria:

```
/home/larry/papers# Is
english-list
history-final
masters-thesis
notes
```

Agora redirecione a saída do comando ls num arquivo chamado arquivolista:

```
/home/larry/papers# Is > arquivo-lista
/home/larry/papers# sort -r arquivo-lista
notes
masters-thesis
history-final
english-list
/home/larry/papers#
```

Assim você gravou a saída do ls num arquivo, e então rodou "sort -r" nesse arquivo. Mas isso é incômodo e usa um arquivo temporário para gravar os dados do ls.

A solução é usar *pipes*. Essa é uma característica do shell que conecta uma série de comandos. A stdout do primeiro comando é enviada para a stdin do segundo comando. Neste caso, queremos mandar a stdout do ls para a stdin do sort. Use o símbolo "|" para criar um pipe, como segue:

```
/home/larry/papers# Is | sort -r
notes
masters-thesis
history-final
english-list
/home/larry/papers#
```

Esse comando é menor e mais fácil de digitar.

Veja outro exemplo útil. O comando: /home/larry/papers# ls /usr/bin

mostra uma longa lista de arquivos, a maioria dos quais voa pela tela tão rápido que você não consegue ler. Assim, vamos usar o more para mostrar uma lista de arquivos em /usr/bin.

/home/larry/papers# ls /usr/bin | more

Agora você pode paginar a lista de arquivos como quiser.

Mas a graça não termina aqui! Você pode os pipes em mais de dois comandos seguidos. O comando "head" é um filtro que mostra as primeiras linhas de uma entrada de dados (neste caso, vinda de um pipe). Se você quer mostrar o último arquivo em ordem alfabética do diretório corrente, use comandos como no exemplo:

/home/larry/papers# ls | sort -r | head -1 notes /home/larry/papers#

onde "head -1" mostra a primeira linha da entrada que recebe (neste caso, a lista ordenada em ordem reversa vinda do ls).

11.4 Personalizando seu ambiente

Um shell fornece muitos mecanismos para personalizar seu ambiente de trabalho. Como mencionado acima, um shell é mais que um interpretador de comando - ele também é uma poderosa linguagem de programação. Embora escrever scripts de shell seja um assunto extenso, gostaríamos de introduzir algumas maneiras de como simplificar seu trabalho no sistema Linux pelo uso dessas funções avançadas do shell.

Como mencionado antes, shells diferentes usam sintaxes diferentes para scripts. Por exemplo, o tcsh usa uma sintaxe parecida com a linguagem C, enquanto o Bourne shell usa outro tipo de sintaxe. Nesta seção, não vamos ver muitas diferenças entre os dois, mas vamos assumir que os scripts sejam executados na sintaxe do Bourne shell.

11.4.1 Variáveis do shell e o ambiente

Um shell permite que você define **variáveis**, como a maioria das linguagens de programação. Uma variável é somente um dado ao qual é associado um nome. O tcsh, assim como outros shells, usam diferentes mecanismos dos descritos aqui para definir variáveis. Esta seção assume o uso de um uma sintaxe tipo Bourne shell. Veja a página do manual para o tcsh para outros detalhes.

Quando você determina um valor para uma variável (usando o operador "="), você pode acessar a variável precedendo um caracter "\$" ao nome da variável, como demonstrado abaixo:

/home/larry# foo="hello there"

A variável foo recebe o valor "hello there". Você pode se referenciar a esse valor pelo nome da variável precedido por um caracter "\$". Por exemplo, o comando

/home/larry# echo \$foo hello there /home/larry#

produz o mesmo resultado que:
/home/larry# echo "hello there"
hello there
/home/larry#

Essas variáveis são internas ao shell, o que significa que somente o shell pode acessá-las. Isso pode ser útil em scripts shell; se você precisa guardar um nome de arquivo, por exemplo, pode armazená-lo numa variável, como acima. O comando "set" mostra a lista de todas as variáveis definidas no shell.

Entretanto, o shell permite que você **exporte** variáveis para o **ambiente**. O ambiente é o conjunto de variáveis que são acessíveis por todos os comandos que você executa. Uma vez que você defina uma variável dentro do shell, se você a exportar, ela também fará parte do ambiente. Use o comando "export" para exportar uma variável para o ambiente.

Novamente, aqui há uma diferença entre o bash e tcsh. Se você usa o tcsh, outra sintaxe é usada para especificar variáveis de ambiente (o comando "setenv" é usado). Veja o manual do tcsh para mais informações.

O ambiente é muito importanto para o sistema UNIX. Ele permite que você configure certos comandos definindo variáveis que esses comandos conhecem.

Aqui temos um exemplo rápido. A variável de ambiente PAGER é usada pelo comando "man" e especifica o nome do comando para seu usado pelo "man" para mostrar uma tela de cada vez. Se você setar a variável PAGER para o nome de um comando, o "man" usará esse comando para paginar as telas, ao invés do "more" (que é o padrão).

Atribua "cat" à variável PAGER. Isso faz com que a saída do "man" role pela tela, sem pausa entre as páginas.

/home/larry# PAGER=cat Agora exporte PAGER para o ambiente /home/larry# export PAGER

Tente o comando "man ls". A página do manual deve rolar pela sua tela sem pausa.

Agora, se você atribuir "more" à variável PAGER, o comando "more" será usado para mostrar a página do manual.

/home/larry# PAGER=more

Note que não temos que usar o comando "export" depois de termos mudado o valor de PAGER. Só precisamos exportar uma variável uma vez. Qualquer mudança feita à variável posteriormente será automaticamente propagada para o ambiente.

Freqüentemente é necessário colocar as seqüencias de caracteres entre aspas para previnir que o shell interprete vários caracteres como especiais. Por exemplo, você tem que colocar aspas em strings que contenham "*", "?", espaços, etc. Há muitos outros caracteres que precisam ser protegidos de interpretação do shell. Uma explicação detalhada é descrita no *Bourne Shell Tutorial* da SSC.

As páginas de manual para um comando particular indicam se esse comando usa alguma variável de ambiente. Por exemplo, a página de manual do "man" explica que a variável PAGER é usada para especificar o comando paginador.

Alguns comandos compartilham variáveis de ambiente. Por exemplo, muitos comandos usam a variável de ambiente EDITOR para especificar o editor padrão a ser usado quando um é necessário.

O ambiente é também usado para informações importantes sobre a sessão de login. Um exemplo é a variável de ambiente HOME, que contém o nome do seu diretório home.

/home/larry/papers# echo \$HOME /home/larry

Outra variável interessante é PS1, que define o prompt principal do shell. Por exemplo:

/home/larry# PS1="Your command, please: " Your command, please:

Para definir o prompt como antes (que contém o diretório de trabalho corrente seguido pelo símbolo "#"),

Your command, please: PS1="\w# " /home/larry#

A página de manual do bash descreve a sintaxe usada para definir o prompt.

11.4.1.1 A variável de ambiente PATH

Quando você usa o comando "ls", como o shell sabe onde encontrar o executável "ls"? De fato, o "ls" está no diretório /bin, na maioria dos sistemas. O shell usa a variável de ambiente PATH para localizar o arquivo executável para o comando que você digitou.

Por exemplo, sua variável PATH pode ser definida como: /bin:/usr/bin:/usr/local/bin:.

Esta é uma lista de diretórios para o shell procurar. Cada diretório está separado por um ":". Quando você usa o comando ls, o shell olha primeiramente em "/bin/ls", e então em "/usr/bin/ls", e assim por diante.

Note que o PATH não tem nada a ver com arquivos regulares (de texto, por exemplo). Por exemplo, se você usa o comando:

/home/larry# cp foo bar

o shell não usa o PATH para localizar os arquivos foo e bar - esses nomes de arquivos são assumidos como caminhos completos. O shell só usa o PATH para localizar o executável "cp".

Isso economiza seu tempo, e você não precisa estar lembrando onde todos os executáveis estão armazenados. Em muitos sistemas, os executáveis estão espalhados em muitos lugares, como em /usr/bin, /bin ou /usr/local/bin. Ao invés de dar o caminho completo para o comando (como /usr/bin/cp), você pode definir o PATH como uma lista de diretórios onde você quer que o shell procure automaticamente.

Note que o PATH contém ".", o que significa o diretório de trabalho corrente. Isso permite que você crie scripts de shell ou programas e rode-os como um comando do seu diretório de trabalho corrente, sem ter que especificá-lo diretamente (como em "./makebook"). Se um diretório não está no seu PATH, o shell não vai procurar nesse diretório por comandos que devem ser rodados; essa regra inclui também o diretório de trabalho corrente.

11.4.2 Scripts de inicialização do Shell

Além de scripts de shell que você cria, há uma série de scripts que o shell usa para certos propósitos. Os mais importantes são os **scripts de inicialização**, que são scripts executados pelo shell quando você dá login.

Os scripts de inicialização são bem simples. Entretanto, eles inicializam seu ambiente executando comandos automaticamente quando você loga. Se você sempre usa o comando "mail" para verificar seus e-mails quando entra, você pode colocar esse comando no script de inicialização para executá-lo automaticamente.

Tanto o bash quanto o tcsh fazem distinção entre um **login shell** e outras invocações do shell. Um login shell é o shell chamado quando você entra no sistema. Usualmente é o único shell que você usa. Entretanto, se você chama um shell de dentro do vi, por exemplo, você inicia uma nova instância do shell, que não é seu shell de login. Além disso, cada vez que você roda um script de shell, você automaticamente inicia outra instância do shell para executar o script.

Os arquivos de inicialização usados pelo bash são: /etc/profile (configurado pelo administrador do sistema e executado por todos os usuários do bash no momento do login), \$HOME/.bash_profile (executado por uma sessão de login do bash), e \$HOME/.bashrc (executado por todas as instâncias do bash que não sejam de login). Se o .bash_profile não está presente, o ".profile" é usado no lugar.

O tcsh usa os seguintes scripts de inicialização: /etc/csh.login (executado por todos os usuários do tcsh no momento do login), \$HOME/.tcshrc (executado no momento do login e a cada nova instância do

tcsh), e \$HOME/.login (executado no momento do login, após o .tcshrc). Se o ".tcshrc" não está presente, o ".cshrc" é usado no lugar.

Um guia completo de programação shell estaria além do escopo deste livro. Veja as páginas do manual para o bash e o tcsh para aprender mais sobre personalizar o ambiente Linux.

12 CONTROLE DE TAREFAS

12.1 Tarefas e Processos

Controle de Tarefas (jobs) é uma característica oferecida por vários shells (incluindo o bash e o tcsh) que permite que você controle múltiplos comandos que estão rodando (as tarefas) ao mesmo tempo. Antes de aprofundar-nos mais, precisamos falar sobre processos.

Cada vez que você roda um programa, você inicia o que é chamado de um **processo**. O comando "ps" mostra uma lista de processos rodando no momento, como mostrado aqui.

/home/larry# ps
PID TT STAT TIME COMMAND
24 3 S 0:03 (bash)
161 3 R 0:00 ps
/home/larry#

O PID listado na primeira coluna é o que chamamos de **ID do processo** (ou identificação), um número único dado a cada processo que está rodando. A última coluna, COMMAND, é o nome do comando que está rodando. Aqui, estamos vendo somente os processos que o Larry está rodando (há muitos outros processos roddando no sistema também - "ps -aux" lista todos). Eles são o bash (o shell do Larry), e o próprio comando "ps". Como você pode ver, bash está rodando concorrentemente com o comando ps. O bash executou o ps quando o Larry digitou o comando. Depois que o ps terminar de rodar (somente depois que a tabela de processos é mostrada), o controle é retornado ao processo bash, que mostra o prompt, pronto para outro comando.

Um processo rodando é também chamado de uma **tarefa**. Os termos processos e tarefas são intercambiáveis. Entretanto, um processo é usualmente referido com o uma terefa quando usado em conjunto com **controle de tarefas** - uma característica do shell que permite chavear entre várias tarefas independentes.

Na maioria dos casos, os usuários rodam somente uma tarefa de cada vez - qualquer comando que eles tenha digitado no shell. Entretanto, usando controle de tarefas, você pode rodar várias tarefas de uma só vez, e chavear entre elas quando necessário.

Como isto pode ser útil? Vamos assumir que você esteja editando um arquivo texto e quer interromper sua edição para fazer alguma outra coisa. Com controle de tarefas, você pode suspender temporariamente o editor, voltar para o prompt do shell e iniciar algum outro trabalho. Quando terminar, você pode chavear novamente para o editor e voltar para onde estava, como se não tivesse saído dele. Há muitos outros usos práticos do controle de tarefas.

12.2 Foreground and background

As tarefas podem ser em **foreground** (primeiro plano) ou em **background** (segundo plano). Só pode haver uma tarefa em primeiro plano por vez. A tarefa em primeiro plano é aquela com a qual você interage - ela recebe entradas do teclado e envia saída para a sua tela, a menos, é claro, que você tenha redirecionado a entrada ou saída, como descrito anteriormente. Por outro lado, tarefas em segundo plano não recebem entradas do terminal - em geral, elas rodam em silêncio sem necessidade de interação.

Algumas tarefas levam um bom tempo para terminar e não fazem nada interessante enquanto rodao. Compilação de programas é uma dessas tarefas, assim como compactar um grande arquivo. Não há razões para que você fique sentado esperando que essas tarefas terminem; basta rodá-las em segundo plano. Enquanto as tarefas rodam em segundo plano, você está livre para rodar outros programas.

Tarefas também podem ser **suspensas**. Uma tarefa suspensa é uma tarefa que está temporariamente parada. Depois que você suspende uma tarefa, você pode solicitar que ela continue em primeiro ou segundo plano, conforme necessário. Resumir uma tarefa suspensa não muda o estado da tarefa de nenhuma forma - a tarefa continua rodando de onde ela parou.

Suspender uma tarefa não é igual a interromper uma tarefa. Quando você **interrompe** um processo rodando (pressionando a tecla de interrupção, que é usualmente o Ctrl-C), o processo é morto, para sempre. Uma vez que a tarefa é abortada, não há como retornar para ela. Você terá que rodar o comando novamente. Além disso, alguns programas capturam a interrupção, e pressionando Ctrl-C não vai abortar imediatamente a tarefa. Isso permite que o programa execute funções para sair de forma limpa antes de terminar. Ainda, alguns programas não permitem que você os aborte com uma interrupção.

12.3 Colocando tarefas em segundo plano e abortando tarefas

Vamos começar com um exemplo simples. O comando "yes" é aparentemente um comando inútil que manda uma lista sem fim de "y"s para a saída padrão. (Na verdade ele é útil se você fizer um pipe com ele na saída de outro programa que pergunta uma série de perguntas "yes" ou "no" e você quer confirmar todas elas).

Tente:
/home/larry# yes
y
y
y
y
y
y

У

Os y's vão continuar infinitamente. Você pode matar o processo pressionando a tecla de interrupção (Ctrl-C). Para que você não tenha que lidar com uma lista inoportuna de y's, vamos redirecionar a saída padrão do "yes" para /dev/null. Como você pode lembrar, /dev/null age como um "buraco negro" para os dados. Qualquer dado mandado para /dev/null desaparece. Esse é um método efetivo de calar um programa que "fala" muito.

/home/larry# yes > /dev/null

Ah, bem melhor! Nada é impresso, mas o prompt do shell não volta. Isso porque o "yes" ainda está rodando, e mandando uma monte desses y's inúteis para o /dev/null. Novamente, para matar a tarefa, pressione a tecla de interrupção.

Vamos supor que você queira que o comando "yes" continue rodando, mas quer ter o prompt do shell de volta para que possa continuar trabalhando. Você pode colocar o "yes" em segundo plano, permitindo que ele rode e sem que você tenha que interagir.

Uma maneira de colocar o processo em segundo plano é anexar um caracter "&" no fim da linha de comando.

```
/home/larry# yes > /dev/null &
[1] 164
/home/larry#
```

Como você pode ver, o prompt do shell retornou. Mas o que é esse "[1] 164"? E o comando yes está realmente rodando?

O "[1]" representa o número da tarefa para processo yes. O shell determina um número de tarefa para cada tarefa rodando. Como o yes é o primeiro e único arquivo que nós estamos rodando, ele recebe o número de tarefa 1. O "164" é o ID do processo, ou PID (process id), número dado pelo sistema para a tarefa. Você pode usar qualquer dos dois números para fazer referência à tarefa, como vamos ver mais tarde.

Agora você tem o processo "yes" rodando em segundo plano, mandando continuamente uma lista de y's para o /dev/null. Para verificar o estado do processo, use o comando interno do shell, "jobs".

```
/home/larry# jobs
[1]+ Running yes > /dev/null &
/home/larry#
```

Com certeza, lá estará ele. Você também pode usar o comando ps, como demonstrado acima para verificar o estado da tarefa.

Para matar a tarefa, use o comando kill. Esse comando recebe o número da tarefa ou o ID do process como um argumento. Esse era a tarefa de número 1, assim, usando o comando:

```
/home/larry# kill %1
```

mata a tarefa. Quando identificar uma tarefa pelo número de tarefa, você deve usar o caracter de percentagem (%) como prefixo ao número.

Agora que você matou a tarefa, use o comando jobs novamente para verificar seu estado.

/home/larry# jobs

[1]+ Terminated yes > /dev/null &

/home/larry#

A tarefa está, de fato, morta, e se você usar o comando jobs ainda mais uma vez, nada deve ser mostrado.

Você também pode matar uma tarefa usando o PID do processo, mostrado junto com o número da tarefa quando você a iniciou. Em nosso exemplo, o ID do processo é 164, então o comando:

/home/larry# kill 164

é equivalente ao

/home/larry# kill %1

Você não deve usar o caracter "%" quando fizer referência a uma tarefa pelo ID do processo.

12.4 Parando reiniciando tarefas

Há outra forma de colocar uma tarefa em segundo plano. Você pode iniciar uma tarefa normalmente (em primeiro plano), **parar** o processo, e então reiniciá-lo em segundo plano.

Primeiro, inicie o processo yes em primeiro plano, como você fez antes:

/home/larry# yes > /dev/null

Novamente, como o yes está rodando em primeiro plano, você não vai receber o prompt do shell de volta.

Agora, ao invés de interromper o processo com Ctrl-C, **suspenda** a tarefa. Suspender uma tarefa não a aborta. Somente a pára temporariamente até que você a reinicie. Para fazer isso, pressione a tecla de suspender, que normalmente é Ctrl-Z.

/home/larry# yes > /dev/null

{Ctrl-Z}

[1]+ Stopped yes > /dev/null

/home/larry#

Enquanto a tarefa estiver suspensa, ela simplesmente não está rodando. Nenhum tempo de CPU é usado para a tarefa. No entanto, você pode reiniciála, o que causa que a tarefa continue rodando, como se nada tivesse acontecido. Ela continuara de onde havia parado.

Para reiniciar uma tarefa em primeiro plano, use o comando "fg" (do inglês, *foreground*).

/home/larry# fg yes > /dev/null

O shell mostra o nome do comando novamente para que você saiba qual tarefa você colocou em primeiro plano. Pare a tarefa novamente com Ctrl-Z. Desta vez, use o comando "bg" para colocar a tarefa em segundo plano. Isso faz com que o comando rode como se você o tivesse iniciado com o caracter "&" (ver seção anterior).

/home/larry# bg [1]+ yes > /dev/null & /home/larry#

E você vai ter o prompt de volta. O comando "jobs" também deve mostrar que o "yes" está rodando, e você pode matar a tarefa com o kill, como fez antes.

Como você pode parar a tarefa novamente? Usar o Ctrl-Z não vai funcionar, pois a tarefa está em segundo plano. A resposta é colocar a tarefa em primeiro plano com fg, e só então pará-la. Como você deve ter notado, pode usar o fg tanto em processos parados como em processos em segundo plano.

Essa é uma grande diferença entre uma tarefa em segundo plano e uma tarefa que está parada. Uma tarefa parada não está rodando - ela não está usando nenhum tempo de CPU, e não está fazendo qualquer trabalho (a tarefa ainda ocupa memória do sistema, embora ele pode ter sido jogado para swap no disco). Uma tarefa em segundo plano está rodando e usando memória, bem como completando algum serviço enquanto você faz outro trabalho.

Entretanto, uma tarefa em segundo plano pode tentar exibir texto em seu terminal, o que pode ser incômodo se você está tentando trabalhar em alguma outra coisa. Por exemplo, se você usar o comando:

/home/larry# yes &

sem redirecionar a stdout para /dev/null, uma lista de y's será mostrada na sua tela, sem que você tenha uma maneira de interrompê-la. (Você não pode usar o Ctrl-C para interromper tarefas em segundo plano). Para parar essa lista infinita de y's, use o comando "fg" para trazer a tarefa para primeiro plano, e depois use o Ctrl-C para matá-la.

Outra nota: os comandos fg e bg normalmente afetam as a última tarefa parada (indicada por um "+" seguindo o número da tarefa quando você usa o comando jobs). Se você estiver rodando múltiplas tarefas de uma vez, você pode colocar tarefas em primeiro ou segundo plano fornecendo o número da tarefa como argumento aos comandos fg ou bg, com no exemplo:

/home/larry# fg %2

(coloca a tarefa 2 em primeiro plano), ou /home/larry# bg %3

(coloca a tarefa 3 em segundo plano). Você não pode usar o ID do processo com fg ou bg.

Além disto, usando o número da tarefa somente, como em: /home/larry# %2

é equivalente a /home/larry# fg %2

Lembre-se que o controle de tarefas é uma função do shell. Os comandos fg, bg e jobs são internos ao shell. Se, por alguma razão, você usa um shell que não suporta controle de tarefas, não espere encontrar esses comandos disponíveis.

Há, também, alguns aspectos que diferem entre os controles de tarefas do bash e tcsh. De fato, alguns shell não fornecem controle de tarefas - entretanto, a maioria dos shells disponíveis para Linux fornece.

13 OBTENDO E INSTALANDO O LINUX

David Bandel reescreveu e revisou a primeira seção sobre instalação do Linux. Partes do trabalho dos seguintes autores de diferentes seções de distribuições Linux também foram adicionadas à primeira versão.

Boris Beletsky escreveu a seção Debian; Sea n Dreilinger, a Slackware; Henry Pierce, a Red Hat; Evan Leibovitch a seção Caldera OpenLinux e Larry Ayers a seção S.u.S.E. Linux.

13.1 Instalação Genérica

Ao contrário de outros sistemas operacionais, o Linux pode ser obtido sem nenhum custo. Como é distribuído protegido pela GNU GPL, não são vendidas licenças do software. Você pode usar Linux sem custo e recomendamos que ajude na sua distribuição.

Isto não significa que empresas não possam lucrar com a distribuição do Linux. Elas podem inclusive adicionar outros software, que não são livres, junto em uma distribuição.

Isto Ihe dá a liberdade de escolha. Se você não pode pagar por um CD-ROM, pode simplesmente copiar de um amigo ou fazer um download da Internet. Tanto comprando de um distribuidor ou fazendo um download de um site de FTP, você vai ter o mesmo sistema operacional rodando e os pacotes oferecidos. Na verdade, você pode conseguir mais softwares livres de um site de FTP do que através de empresas que distribuem em CD devido às restrições que alguns autores impõem à distribuição de seus softwares.

13.1.1 Principais distribuições Linux

Uma visão aprofundada de algumas distribuições Linux inicia na seção "Distribuições Linux". Essas distribuições são: Debian, Red Hat, Caldera, Slackware e S.u.S.E. Cada seção tem informação sobre onde conseguir cada distribuição. Mas lembre-se, o kernel é Linux. O sofware é parte da distribuição, não do Linux. A maior parte do software está disponível livremente e pode ser portada para vários outros UNIX. A principal diferença está no suporte de várias bibliotecas (software chamado por cada aplicação).

Cada distribuição tem suas próprias ferramentas de instalação e manutenção que ajudam na instalação e manutenção do sistema. Cada uma é aparentemente voltada para um público diferente. Com qualquer uma você pode instalar e rodar o Linux. Assim, eu recomendo que você leia sobre cada uma delas e toque idéias com amigos experimentados. Normalmente, grandes cidades têm grupos de Linux, a maioria com usuários experientes, que defendem por que cada distribuição é melhor. Eu sugiro que você escute alguns argumentos e decida. Você também pode consultar listas de e-mail (prefira assinar uma de cada vez) e ler o que outros usuários comentaram e as respostas dos gurus Linux. Assim como as distribuições são bem diferentes, também são variadas as listas de discussão. Parar para fazer a melhor escolha é importante, porque trocar a distribuição normalmente significa reinstalar tudo do zero.

13.1.2 Planejamento básico

Esta seção assume que o novo usuário Linux:

- tem um computador com MS-DOS e Windows ou OS/2;
- tenha um entendimento básico do MS-DOS, mas não do Linux;
- saiba ou possa vir a saber que tipo de hardware está instalado no seu computador;
- tenha desejo de "tentar" usar o Linux, por qualquer razão, nem sempre para mudar o sistema definitivamente (ainda), e
- não tenha uma máquina sobrando ou outro disco rígido, mas várias centenas de MegaBytes livres para uso.

Estas suposições não são extremas e podem até ser um pouco conservadoras. Alguns dizem que se seu vídeo cassete pisca 12:00, Linux não é para você, mas isto me deixaria do lado de fora também. Meu vídeo cassete pisca 12:00.

Antes de iniciarmos, devemos saber aonde queremos chegar. Ainda que seja possível chegar de Nova Iorque a California tentando qualquer via aleatoriamente, a maioria optaria em seguir a rota mais direta. Da mesma forma para instalar o Linux.

13.1.3 Hardware

Esta seção explica os passos necessários antes da instalação. Cada distribuição cobre esta preparação de forma levemente diferente. Ainda que as instalações pareçam diferentes, elas satisfazem às mesmas necessidades e têm mais em comum do que pode parecer. Todas requerem:

- planejamento;
- obter informações do hardware do sistema;
- fazer cópia de segurança do sistema antigo (opcional, mas fortemente recomendado):
- preparar partições Linux;
- decidir qual boot loader instalar (para sistemas com dual boot);
- dar boot no kernel do Linux:
- instalar o kernel:
- escolher e instalar pacotes de software;
- carregar o software;
- fazer a configuração final e ajustes; e
- reiniciar e rodar o sistema.

Agora que descompliquei suficientemente o processo, vamos aprofundar-nos nas etapas. Espere antes de instalar. Não é má idéia aprender com os erros dos outros.

13.1.4 Planejamento

Não vamos exagerar nesta etapa. Qualquer piloto diria que a aterrisagem vai depender da aproximação. O mesmo vale para o Linux.

Primeiramente temos que determinar que tipo de hardware você tem. Um checklist foi incluído para assisti-lo. Seja tão preciso quanto possível, sem exageros. Por exemplo, se você tem uma placa de rede, você precisa saber qual tipo (ex: SMC-Ultra, 3 Com 3C509, etc), endereço de I/O (ex: io=0x300), interrupção (IRQ 10), mas não o endereço de hardware (ex: 00 00 a6 27 bf 3c). Nem toda a informação sobre o hardware é necessária. Se você tem Windows 95 ou NT rodando, você pode conseguir alguns dados pela janela de "Propriedades do Sistema". Fora isso, consulte os manuais de hardware ou o site das empresas fabricantes. Como é muito importante, vamos revisar esta planilha.

13.1.5 Planilha para planejamento do sistema

Os campos sublinhados, na planilha abaixo, são exemplos de uma configuração hipotética. Os campos entre parênteses são exemplos de respostas válidas.

Planilha Geral:

Processador:

Tipo: 386 486 Pentium PPro

Velocidade (opcional):

Fabricante: Intel/AMD/Cyrix/...

Placa Mãe:

Fabricante: desconhecido

Chip Set:: triton II

Mouse:
Fabricante:

Tipo: bus PS/2 porta serial

Interface serial: COM1 (ttyS0), COM2 (ttyS1)

HD(s):

Tipo: <u>IDE/MFM/RLL/ESDI/SCSI</u> Tamanho (liste cada disco): **Controlador SCSI (se houver):**

Fabricante: BusLogic

Modelo: 948

Boot: (Linux - DOS/Windows - OS/2 - Outro)

Disco: Partição: Tamanho: Boot: Partição: Disco: Tamanho: Boot: Disco: Partição: Tamanho: Boot: Disco: Partição: Tamanho: Boot:

CD-ROM: (IDE/ATAPI - SCSI - Outro)

Fabricante:

Modelo:

X-Windows:

Placa de vídeo:

Mfg:

Model:

Memória da placa: (8Mb/16Mb/32Mb/64Mb)

Monitor: Fabricante:

Modelo:

Max taxa de refresh:

Rede: Modem: Fabricante: Modelo:

Porta serial: COM1: ttyS0 COM2: ttyS1 COM3: ttyS2 COM4: ttyS3

Nome do computador: Aguia

As próximas respostas são necessárias somente se você estiver usando uma placa de rede (NIC - network interface card). Não configure a rede se você não tiver uma placa de rede instalada.

Tipo de NIC: (Ethernet - Token Ring - FDDI - outro)

Fabricante do NIC:

Modelo:

Nome do domínio da rede: aves.net

Endereço IP: <u>192.168.1.2</u>

Endereço de rede: <u>192.168.1.0</u>

Máscara: <u>255.255.255.0</u>

Endereço de Broadcast: 192.168.1.255

Gateway(s): (none - 192.168.1.1)

DNS(s): <u>192.168.1.2</u>

Algumas das informações, na seção *Planilha Geral,* foram colocadas para referência futura somente. Especificamente, não precisamos saber agora o tipo de processador. Também podemos continuar sem saber o chip set da placa mãe. Mas se a informação existir, é bom que a tenhamos.

13.1.6 Mouse

Outras informações, como o tipo de mouse, são necessárias (se você quiser usar o mouse, claro). Vamos precisar conhecer o fabricante do mouse porque há diferenças na implementação de como os sinais são tratados. Aqui, atenção ao detalhe é essencial. Se você tem um mouse Microsoft, ele ainda pode ter uma interface serial ou PS/2. Observando o conector atrás do computador pode não ser suficiente. Vários computadores vêm com um mouse que parece serial, mas estão conectados internamente a uma porta PS/2 na placa mãe.

Leia o que está impressa em baixo do mouse cuidadosamente antes de decidir. Observe também se há uma chavezinha na parte de baixo, dizendo Micrososft / PC systems. Nesse caso, prefira PC systems. A configuração Microsoft não suporta o botão do meio, que pode ser útil no UNIX.

Talvez você precise saber outra informação adicional que não listamos acima: o dispositivo (*device*) pelo qual o sistema acessa o mouse. O Linux deve saber como o *device* será chamado. Se você tem um PS/2, provavelmente usará <code>/dev/psaux</code>, ou ainda <code>/dev/psmouse</code>. Para mouses seriais, se você conhece as portas COM1: e COM2: do MS-DOS, substitua-as por <code>/dev/ttys0</code> e <code>/dev/ttys1</code> respectivamente. Vou evitar explicar ar origens do nome <code>tty</code>, pois isso consumiria vários parágrafos e já é coberto em outros livros UNIX.

13.1.7 Discos Rígidos (HDs) e CD-ROMs

Antes de começar a instalação, você precisa decidir quanto espaço do seu disco você vai dedicar ao Linux e saber de quanto dispõe. Decidindo durante a instalação você estará procurando por problemas e acabará perdendo mais tempo, talvez dados e ainda terá que reinstalar tudo.

Seu disco pode ser de vários tipos. Para nossos propósitos, IDE, MFM, RLL e ESDI são equivalentes, e eu usarei somente o termo IDE. Este também engloba o EIDE, a interface mais comum em uso hoje em dia, favorecida pelo baixo custo.

Se seu disco tem interface SCSI, será evidente durante o boot. Você deverá conhecer o fabricante e modelo do controlador SCSI. Os controladores mais comuns são Adaptec e BusLogic, mas não são os únicos. Estes também têm modelos específicos, como AHA-1572 ou BTC-958. Esta informação é freqüentemente mostrada durante a inicialização.

Para reservar espaço, precisamos saber o tamanho do disco rígido. Se você tem MS-DOS ou Windows, ou OS/2 no seu computador, o Linux deve ter uma partição separada. Ele pode ser carregado numa partição MS-DOS com UMSDOS, mas não falaremos sobre isso. O Linux tem emuladores DOS que pode ler e até mesmo rodar alguns programas DOS e o DOS usualmente pode "ver" o que há numa partição Linux.

Se você tem e quer manter o MS-DOS (assume-se), precisamos determinar quanto espaço vai ser reservado para ele. Subtraia este número do tamanho total do disco e guarde o restante. Por agora, anote o tamanho total do disco e o segundo número com o tamanho total que dedicará ao Linux.

Para um drive de CD-ROM você vai precisar informações similares. Ele pode ser IDE/ATAPI (o mais comum em sistemas atuais), SCSI, ou um antigo, com controlador proprietário, como esses que vinham com placas de som. Se você tiver um IDE ou SCSI, melhor. Se tiver um proprietário, precisará conhecer o fabricante e modelo para que o Linux possa reconhecê-lo.

13.1.8 Discos no Linux

Para novatos no Linux que estão familiarizados somente com MS-DOS, e para aqueles que vêm de plataformas UNIX, dispositivos (*devices*) no Linux têm referências peculiares. Estas referências são usadas quase desde o início e entendê-las um pouco é necessário.

No Linux, assim como em qualquer UNIX, devices são arquivos especiais. Discos rígidos são tratados como arquivos e referenciados por nomes, assim como os modens, monitores, e outros dispositivos de hardware. UNIX trata os dispositivos como arquivos de onde se lê e grava informação. Como o Linux os vê como arquivos, eles estarão localizados em um diretório dedicado aos dispositivos. Após a instalação você poderá vê-los sob o diretório /dev, que vem do inglês, devices.

Mesmo vistos pelo Linux como arquivos, eles são especiais. Podem vir em dois "sabores" (tipos): **block** ou **character**, que fazem referência ao modo como o dispositivo se comunica: em blocos de dados ou caracteres individuais. Os dispositivos são criados automaticamente durante a instalação.

(.....)

13.1.9 Instalando o X Window

Incluí na planilha informações sobre sua placa de vídeo e monitor. Mesmo não sendo absolutamente necessário, muitos dos que vêm do mundo Windows querem instalar e configurar uma interface gráfica (GUI - graphical user interface). Algumas distribuições já fazem essa instalação, outras fornecem pacotes para instalar posteriormente. Nesse caso, a informação sobre o seu hardware será importante.

Você deverá saber o fabricante e modelo específico de sua placa de vídeo. Algumas placas podem ser identificadas por software, outras não. De qualquer forma, conhecer quanta memória RAM tem a placa e o chip set usado (ex: S3, S3-Virge), é importante. Esta informação economiza muito tempo e sofrimento. A parte mais difícil e frustrante de qualquer instalação Linux é a instalação do X Window.

A informação do seu monitor pode ser um pouco mais difícil de obter. Se você tem um uma marca desconhecida, pode ter de fornecer as taxas de refresh horizontal e vertical por conta própria.

Se estiver em dúvida, seja conservador. Se você usar uma taxa maior do que seu monitor suporta, pode queimar o monitor ou a placa de vídeo.

Sobre o mouse, outro dispositivo que o X usa, já recolhemos a informação necessária na seção 2.1.7.

13.1.10 Hardware de rede

Esta seção não é tão significante quanto a planilha pode sugerir. (.....). No entanto, se você tem uma placa de rede (NIC), seja ela Ethernet, token ring ou outra, deve conhecê-la antes de continuar. Esta informação será necessária durante a instalação para que você possa usar a placa no futuro.

Durante a instalação inicial do Linux você não precisa ter um NIC. Você pode pular algumas partes sobre rede. No entanto, todos os computadores devem ter um nome no Linux. O exemplo na planilha assume que você tenha escolhido um assunto, *aves*, por exemplo, e você vai nomear os computadores nesse grupo, mas qualquer esquema que você inventar está ok.

Se você tem um modem, deve saber onde ele está conectado. As portas seriais, /dev/ttys0 até /dev/ttys3, correspondem às portas COM: 1 até 4 do MS-DOS ou Windows.

Isto termina nossa planilha sobre a metade do planejamento que precisamos fazer. Note-se que não está na nossa planilha o total de memória RAM que o sistema tem. Felizmente, o Linux pode rodar com sistemas que tenham até menos de 4MB de RAM, mas isto vai ser penoso para a instalação e qualquer uso subseqüente. Se você tem 4MB de RAM ou menos, deverá seguir procedimentos especiais para máquinas com pouca memória.

13.1.11 Planejamento, Parte 2

Partes da seção seguinte, particularmente sobre estratégias de particionamento, são bem controversas para instaladores experientes, mas eu darei minhas dicas. Se você se sentir à vontade, poderá adotar seus próprios critérios. Creio que a maior parte das diferenças de opinião advém dos diferentes propósitos para o sistema, ex: uma estação de trabalho, um servidor Web, um servidor de News, ou outra função.

13.1.12 Estratégias de particionamento

Poucos usuários experientes do Linux vão aconselhá-lo a criar uma partição Linux nativa e uma partição de swap e iniciar a instalação. Há muitos motivos para isso, e eu compartilho com eles. Assim, eu crio várias partições nativas Linux. Para mim, a razão mais convincente é que um dia você vai querer fazer um upgrade no sistema, o que vai requerer uma reformatação do sistema de arquivos. A distribuição Slackware, por exemplo, não permito um upgrade do sistema instalado, e não parece que terá no futuro. No upgrade do kernel 0.99 para o 1.2.13 tive que reformatar tudo, e assim também foi no upgrade do 1.2.13 para o 2.0.0. Certamente nunca vou querer perder os arquivos que acumulei no meu diretório home. Claro que tenho uma cópia de segurança, mas mantendo meu /home intacto é mais fácil, especialmente se eu tenho todos os meus arquivos especiais em um sudiretório lá.

Outra razão é que cada partição de boot deve estar dentro dos primeiros 1024 cilindros do disco. Quando um PC inicializa, uma seqüencia de eventos ocorre que terminam com o carregamento do sistema operacional. Devido a limitações na BIOS (Basic Input/Output System), até que o sistema seja carregado, somente os primeiros 1024 cilindros do primeiro ou segundo disco

rígido podem ser acessados.

Para sentir do que estamos falando, vou descrever um sistema de arquivos Linux padrão e como o Linux lida com partições.

No MS-DOS ou Windows, cada partição é um drive diferente, e pouca distinção é feita se se trata de um dispositivo físico ou lógico (partição). No Linux, dispositivos físicos e lógicos são muito menos rigidamente definidos.

Durante a instalação, você deve escolher qual partição será sua partição root (principal ou raiz). A partição root é definida com "/". Quando nos referenciamos a "/dev" estamos falando de dois diretórios, "/" e "dev". O kernel do Linux estará localizado na partição root, podenso ser num subdiretório, desdee que este subdiretório esteja na partição root. Algumas distribuições, por exemplo, usam /boot para guardar o kernel e outros arquivos de inicialização.

A seguinte estrutura (no mínimo) será criada na sua partição root durante a instalação:

/bin

/dev

/etc

/home

/lib

/lost+found

/proc

/root

/sbin

/usr

/var

Você poderá encontrar outras, como /boot, /mnt, /cdrom, /floppy, /opt, etc, mas as mostradas acima são essenciais.

E sobre outras partições? O Linux pode usar um nome de diretório (/usr, por exemplo), como um **mount point** (ponto de montagem). Isso significa que a outra partição no disco (ou mesmo em outro disco) está montada, neste caso, em /usr.

Se você desmonta uma partição e olha no subdiretório que o Linux usa como ponto de montagem, você vai (ou deveria) vê-lo vazio, sem arquivos ou outros diretórios. Quando a partição é montada, você vai ver os arquivos e diretórios que estão naquela partição. Assim, se você tem dois discos, um com 120MB e outro com 840MB, você pode fazer uma partição no de 120MB (pode ser a partição root) e montar quaisquer partições que você tenha criado no de 840MB (pode ser uma só ou várias pequenas partições) nos respectivos pontos de montagem, uma partição por ponto de montagem, criando, efetivamente, um sistema com 960MB.

A restrição é que você não pode usar certos diretórios, que estão no disco root, como pontos de montagem, pois eles contêm arquivos que são necessários ao boot ou para montar outros sistemas. Obviamente se o comando usado para montar outras partições está localizado em outra partição

e você não pode acessar essa partição até que você a tenha montado, você estará como um cão correndo atrás do seu rabo.

Os diretórios que não podem ser usados como pontos de montagem São: /bin, /dev, /etc, /lib, /lost+found, /proc, /root, @ /sbin.

(.....)

Vamos olhar um pequeno exemplo. Você quer montar um provedor de acesso à Internet (ISP - Internet Service Provider). Você tem quatro máquinas, cada uma com um 1GB de disco. Você decide, então, dividir o espaço da seguinte forma:

máquina A: /= 120MB
/usr = resto do disco (exportado)
/home = 0 - montado da máquina B
/var/news = 0 - montado da máquina C
/var/spool/mail = 0 - montado da máquina D

máquina B: /= 120MB
/usr = 0 - montado da máquina A
/home = resto do drive (exportado)
/var/news = 0 - montado da máquina C
/var/spool/mail = 0 - montado da máquina D

máquina C: /= 120MB
/usr = 0 - montado da máquina A
/home = 0 - montado da máquina B
/var/news = resto do disco (exportado)
/var/spool/mail = 0 - montado da máquina D

máquina D: (exercício - monte a tabela)

Você provavelmente notou que eu arbitrariamente reservei 120MB para a partição de root e defini o resto para as outras (/usr, /home, /var/spool/mail e assim por diante). Eu também não reservei qualquer espaço para uma partição de swap. Assim, vamos dar uma olhada no que vamos precisar, tendo em mente que "depende do caso". Eu vou abordar este assunto na perspectiva de um usuário caseiro, com poucos usuários, sem outras necessidades extraordinárias.

A melhor maneira de começar é descrevendo como meu computador é. Eu tenho dois discos, /dev/hda (1.2 GB) e /dev/hdb (540 MB). O comando df (disk free) mostra:

File system	1024-bloc	ks Used	Available	Capacity	Mounted on
/dev/hda1	150259	69605	72894	49%	1
/dev/hda3	723923	615452	71075	90%	/usr
/dev/hda2	150291	93326	49204	65%	/usr/X11R6
/dev/hdb1	499620	455044	18773	96%	/home

Você pode notar que eu tenho uma partição root de 150MB com 50% de uso, uma partição /usr quase cheia, uma partição /usr/x11R6 bastante usada, e uma partição grande de 500MB apertada onde está o /home. O resto do disco hdb é uma partição de swap, que não aparece no df.

No mínimo, eu sugeriria reservar uns 100MB para sua partição root, calcular quantos MB você vai precisar por usuário para sua partição /home, reservando este espaço, uns 100 ou 128MB para swap (ver próxima seção), e o resto para /usr. Sua partição /usr deverá ter no mínimo uns 250MB, mas isso vai depender do que você vai instalar. Como você poderá ver, isso pode crescer rapidamente com programas, bibliotecas e dados. Lembre-se também que partições dão a flexibilidade que você não tem com uma partição gigante.

13.1.13 A partição de swap

Você deve pensar numa partição de swap. Diferentemente do Windows, o Linux usa uma partição dedicada de swap. Embora seja possível criar um arquivo de swap, isto não é recomendado. O Linux pode usar até 128MB de swap. Eu recomendaria um mínimo de 80MB, mas quanto mais você puder reservar, melhor.

Uma última consideração antes de você decidir como melhor dividir o seu disco. Lembre-se que eu disse que a BIOS não pode ver além do setor 1023 no disco (mais ou menos uns 512MB). Assim, o kernel do Linux (um arquivo com o provável nome vmlinuz no seu disco de boot), ou qualquer outro kernel de um sistema operacional, devem residir integralmente em um dos primeiros dois discos (/dev/hda ou /dev/hdb) e dentro dos primeiros 1024 setores, ou a BIOS não poderá carregá-los. Para assegurar-se de que isso seja possível, planeje para fazer com que sua partição root (assim como qualquer partição de boot) caia inteiramente nesta limitação no primeiro ou segundo disco rígido.

13.1.14 Reparticionamento

No início deste capítulo eu disse que faria algumas poucas suposições. Uma delas é que manteria seu MS-DOS ou Windows. Como provavelmente seu micro só tenha o Windows, não há razão para que tenha mais de uma partição. Dessa forma, seu disco provavelmente está totalmente dedicado ao Windows.

De qualquer forma, teremos dois sistemas operacionais nesse computador. Se atualmente seu disco está vazio (sorte sua), isso é ótimo, mas você ainda não está exatamente pronto para saltar esta etapa. O Linux fica confortável em qualquer lugar que você o instale. A BIOS pode não ser capaz de inicializá-lo, mas uma rodando, ele não reclamará de estar colocado na

quarta partição do quarto disco. O Windows e o MS-DOS não são tão complacentes. Eles querem o primeiro disco e a primeira partição e podem se negar a inicializar em qualquer outra posição. Eu já vi um MS-DOS inicializar no segundo disco, mas o primeiro disco não tinha partições MS-DOS, assim o DOS não o reconheceu. A melhor estratégia usualmente é seguir o caminho que oferece menor resistência. Se possível, instale o Windows na primeira partição do primeiro disco.

Uma segunda consideração em um sistema múltiplo é qual sistema operacional carregar primeiro. Se você estiver tentado a particionar o disco e instalar o Linux primeiro (reservando, por exemplo, /dev/hdal para o Windows), e então instalar o Windows, não faça isso. O Windows é terrivelmente desrespeitoso. Os sistemas operacionais Microsoft, em geral, vão apagar qualquer *boot loader* que você tenha instalado no seu MBR (Master Boot Record - setor do disco que a BIOS usa para apontar para o kernel). O Linux não causa esses problemas e até fornece uma maneira de escolher qual imagem do kernel será carregada. Ele também permite sua intervenção durante o processo para escolher qual sistema operacional deve ser carregado. Esta é uma etapa padrão das instalações Linux.

13.1.15 Fazendo cópia de segurança de seu sistema antigo

Antes de "colocar a mão na massa" na tabela de partições, vamos revisar os procedimentos para proteger seus dados que estão no disco rígido. Estes procedimentos assumem que você tenha uma partição FAT ou FAT32. Outros sistemas operacionais podem ou não ter maneiras de efetuar os mesmos passos.

A primeira coisa a fazer é uma cópia de segurança (backup) completa de todos os dados. As ferramentas que você vai usar na instalação com certeza funcionam, mas os procedimentos são inerentemente perigosos. Em qualquer momento que você esteja trabalhando numa tabela de partições, você pode facilmente perder todos os dados contidos no disco. Faça uma cópia de segurança antes de prosseguir.

Uma vez que você tenha feito o backup, cria um disquete de inicialização para seu sistema. No Painel de Controle do Windows, clique em "Adicionar ou remover programas", depois na guia "disco de inicialização" e em seguida no botão "Criar disco...". Depois, copie o arquivo RESTORRB.EXE da distribuição Linux para esse disco.

Rode um desfragmentador no seu disco. Se o desfragmentador encontrar erros, você precisará passar um SCANDISK para corrigi-los. Depois de desfragmentar e corrigir eventuais erros no disco, você estará pronto para rodar o FIPS.EXE para encolher a partição DOS.

13.1.16 FIPS.EXE

No CD de sua distribuição Linux (ou site Internet), você encontrará uma cópia do FIPS.EXE, que pode encolher uma partição FAT ou FAT32. Se você tem outros tipos de partição para encolher, talvez precise do Partition Magic, mas este é um software comercial e não é grátis. Copie o FIPS.EXE para seu

disquete de inicialização e reinicie o sistema com ele. Com isso estaremos certos de duas coisas: seu drive de disquete funciona e você terá inicializado o sistema em Modo Real, e não rodando o Windows.

No prompt A:>, digite FIPS. Você será saudado e questionado sobre qual disco você quer operar (se tiver mais de um). Selecione o disco para encolher. Uma vez que você confirme a escolha, deixe o FIPS fazer uma cópia de seus setores de root e boot para o disco no caso de algo dar errado.

O FIPS irá perguntar se você deseja que todo o espaço livre na partição deve ser usado para criar uma segunda partição. Se você disser "sim", estará deixando a partição DOS sem espaço livre para salvar qualquer coisa, por isso diga "não". Você, então, poderá alterar o espaço reservado para a primeira e a segunda partições. Note que se você não desfragmentou o disco corretamente, não terá muito para deixar para a segunda partição.

Você pode editar e reeditar a tabela até que esteja satisfeito. Uma vez que goste da distribuição de espaço entre as partições, confirme suas escolhas e grave a tabela.

Uma vez que o FIPS tenha terminado, remova o disquete de inicialização do seu computador. Neste exemplo, vamos destruir e recriar a segunda partição durante a instalação para criar ao menos duas partições para o Linux: uma partição de swap e uma partição Linux nativa. Você pode criar quantas quiser.

13.1.17 Preparando o Linux para inicializar

Para instalar o Linux, devemos começar por inicializar o kernel do Linux. Isto é feito da mesma maneira como você instalaria o Windows: precisamos de um disquete de inicialização. A maior parte das distribuições vem somente com um CD-ROM, e mesmo que você já tenha um Linux rodando, a forma de criar disquetes de inicialização para o Linux é diferente do Windows. Se você tiver um computador com um CD-ROM inicializável (a maioria hoje em dia provavelmente tenha que mudar alguma opção na CMOS), algumas distribuições permitem que você inicie desta maneira. De qualquer forma, vamos passar pelo processo de criar o disquete.

13.1.18 Criando um disquete de inicialização do Linux no DOS

Cada CD de distribuição contém um programa DOS que permite gravar uma imagem pronta de um disquete de inicialização num disquete virgem. Insira o disquete no drive. No CD (ou no seu disco, se você fez download da distribuição), encontre o RAWRITE2.EXE (ou RAWRITE.EXE).

Vá para o diretório onde está a imagem do disquete de inicialização. Pode haver mais de uma, variando para configurações diferentes de hardware. Você terá de consultar a documentação da sua distribuição. Rodando o RAWRITE2.EXE sem argumentos resulta em duas perguntas que você terá de responder: o caminho para encontrar o arquivo com a imagem e o drive de destino (A: ou B:). Para encurtar as perguntas *C:> rawrite diskimage drive*

Utilize um bom disquete de inicialização. Muitas inicializações fracassam por disquetes com defeito. O RAWRITE não verifica erros no disco.

Isso também vale para disquetes criados no Linux. O manual badblocks(1) descreve como checar erros de disco.

Nomeie o disco para uso futuro.

13.1.19 Criando um disquete de inicialização no Linux

Se você tem um sistema operacional Linux e quiser gerar um disquete de inicialização, vá para o diretório que contenha as imagens e rode o seguinte comando:

dd if=diskimage of=dispositivo-de-disquete bs=512 conv=sync; sync

Substitua diskimage pelo nome do arquivo com a imagem e indique corretamente o dispositivo disquete (usualmente /dev/fd0), e repita para cada disquete que você precisar. Os parâmetros do dd são: if para input file (arquivo de entrada); of para output file (arquivo de saída) - aqui usamos o drive de disquete; bs para block size (tamanho do bloco), neste caso 512 bytes; conv=sync assegura que o arquivo de saída é exatamente do mesmo tamanho que o de entrada. O "sync" restante assegura que vamos descarregar todos os buffers para o disquete imediatamente.

Um método alternativo que funciona, mas que é freqüentemente evitado por administradores Linux é o comando cp (copy)

cp diskimage dispositivo-de-disquete; sync

Novamente, substitua diskimage pelo arquivo de imagem e dispositivode-disquete pelo dispositivo correto. Pode aparecer uma mensagem pedindo confirmação de que você quer mesmo substituir o dispositivo de disquete por arquivo de imagem. Obviamente isto não ocorre, uma vez que o disquete não é verdadeiramente um arquivo, mas sim um dispositivo, mas o cp não presta atenção a esse detalhe. Responda "sim", se ele perguntar.

Com os disquetes de inicialização do Linux em mãos, você está pronto para instalar o sistema. A maior parte das distribuições chama o fdisk do Linux, aí você poderá criar a partição Linux nativa e a de swap. Para instalar programas, continue criando o sistema de arquivos (equivalente a formatar um disco com DOS) para o Linux e a partição de swap, inicialize a partição de swap e monte a partição Linux.

Uma pergunta que será feita é se você quer checar seu disco rígido contra blocos defeituosos. Se você estiver usando um SCSI, diga "não". Discos SCSI têm um sistema interno de detecção e correção. IDE e outros discos similares não têm isso e precisam ser verificados. Se você tem um disco mais antigo, você certamente vai querer fazer isso. Se você disser "sim", o programa de instalação irá chamar o programa badblocks para mapear todos os blocos defeituosos. Isso leva algum tempo. Na dúvida, diga "sim".

13.1.20 Particionando o disco rígido: fdisk e cfdisk.

Qualquer sistema operacional, seja ele o DOS, o Windows ou Linux, tem sua versão do fdisk. Se você quer criar uma partição para usar no DOS, use a versão DOS FDISK.EXE para criar a partição e gravar a tabela de

particionamento. Se você vai criar uma partição para o Linux, deve criá-la com a versão Linux, fdisk.

No Linux, dois programas de particionamento estão disponíveis: o original fdisk, e uma versão mais amigável cfdisk. A diferença entre os dois é que no fdisk você faz todos os comandos por teclado, através de letras e números. Com o cfdisk, você pode usar uma seta para marcar as opções e apertar <Enter> para executar o comando. O único momento que você não usa as setas ou o <Enter> é quando você especifica o tamanho da partição.

Essencialmente, para iniciantes, todos os disquetes de inicialização são iguais. Reinicie o computador com o disquete no drive. Você receberá uma mensagem de boas vindas

LILO boot:

e um cursor piscando. Se você usar a tecla <Tab>, poderá ver uma lista de nomes. Os nomes diferentes dependem da distribuição, mas procure um que diga "rescue" ou "expert" e outro "install". O "install" iniciará o programa de instalação após a carga do kernel. Assim, se você quer que o programa de instalação passe pelas etapas de particionamento e inicialização do sistema de arquivos, use o "install", senão, use outra opção. Você pode também passar alguns parâmetros no boot do Linux. Para nossos propósitos, isto não será necessário, mas você logo descobrirá se este for o caso.

Entre uma opção e pressione <Enter>. Quando o kernel do Linux terminar o processo de inicialização, você poderá receber um de vários prompts, dependendo da distribuição. Se você receber um prompt do shell, como um caracter '#' ou um sinal de dólar ('\$'), você está onde deveria. Se não, tente pressionar Alt-F2 ou Alt-Shift-F2. Você deve ser capaz de ativar um console virtual do sistema.

Uma vez que consiga um prompt (provavelmente não terá que fazer um login), você estará trabalhando como "root". Entre o seguinte comando:

fdisk

Se retornar erro, tente cfdisk. Esta é a ferramenta para particionar o disco. Por default, ela usa o disk /dev/hda. Assim, se você precisa trabalhar no segundo disco, use:

fdisk /dev/hdb

No fdisk, pressione m para ver o menu. Os comandos que você vai usar são:

- n para criar uma nova partição
- d para destruir uma partição
- t para mudar o tipo de uma partição (83 é Linux nativo, 82 é swap Linux)
- p para exibir a tabela de partições que está na memória (não o que está no disco)
- w para escrever a tabela de partição que está em memória no disco
- q para sair (quit)

Até que você use o comando w, você não está comprometido e pode fazer todas as mudanças que quiser e sair sem confirmá-las.

Preste atenção aos prefixos e sufixos que modificam o tamanho da partição. Os sufixos "k" ou "M" (não importando se caixa alta ou baixa), especificam se o número está em KB ou MB, respectivamente.

Uma nota final sobre partições: você pode criar até quatro partições primárias. Se você precisa mais de quatro partições, você vai criar três partições primárias e outras partições estendidas. Os números das partições estendidas iniciam em 5, assim você poderá ter /dev/hda1, /dev/hda2, /dev/hda3, /dev/hda5, e /dev/hda6 se precisar de cinco partições.

Antes de gravar a tabela de partições, assegure-se que as mesmas não se sobreponham. Desde que os segmentos de início e fim não se sobreponham a quaisquer outros segmentes, você pode estar certo que os limites das partições estarão ok. Partições que começam após o cilindro 1024 podem estar listadas com esse número. Por agora, somente lembre-se que a BIOS não poderá ler (ou inicializar) dessa partição.

cfdisk faz exatamente a mesma coisa que o fdisk, mas mostra na tela o estado da tabela de partição em memória (não a do disco) durante todo o tempo. Use as setas para cima e para baixo para selecionar em qual partição quer trabalhar e as da direita e esquerda para selecionar a ação a ser executada. Pressione <Enter> para executar a ação. Você terá que entrar números para o tamanho das partições, mas toda a informação necessária estará na tela. cfdisk inicia no /dev/hda, por isso, se quiser trabalhar no segundo disco, use o parâmetro /dev/hdb . Lembre-se de salvar a tabela antes de sair. Esta é a pior parte do cfdisk. Ele não pergunta por confirmação antes de sair. Assim, selecione "Write" e depois pressione <Enter>.

14 DISTRIBUIÇÕES LINUX

Agora você vai enfrentar a tarefa de decidir qual distribuição particular é melhor para as suas necessidades. Nem todas as distribuições são parecidas. Muitas vêm com todos os software necessários para rodar um sistema completo. Outras são distribuições "menores" dirigidas a usuários sem tanto espaço de disco disponível. Muitas contêm apenas o coração do sistema Linux, e você tem que instalar pacotes grandes, como o X Window, por conta própria.

(.....)

Se você tem acesso ao news, você pode pedir opiniões de pessoas que tenham instalado o Linux. Além disso, o Linux Journal mantém uma tabela de características comparando as distribuições e periodicamente publica revisões sobre as mesmas (em http://www.linuxjournal.com você vai encontrar uma versão online e artigos). Se você conhecer alguém que tenha instalado o Linux, tanto melhor. Pergunte a ele e peça ajuda. Há muitos fatores a considerar quando da escolha de uma distribuição; no entanto, as necessidades e opiniões de cada um são diferentes. Atualmente, a maior parte das distribuições contém essencialmente o mesmo conjunto de software. Dessa forma, sua escolha vai ser mais ou menos arbitrária.

14.1 Debian GNU/Linux

Esta seção sobre Debian GNU/Linux foi escrita por Boris Beletsky.

14.1.1 Características da instalação do Debian GNU/Linux

Dependências: sim

Método de inicialização da instalação: disquete

Métodos de instalação: CD, HD, NFS, FTP

Inicialização do sistema: Sys V init Facilidade de instalação: desafiadora

Ferramentas gráficas de administração: não Ferramenta de instalação: dselect

Ferramenta para manutenção de pacotes: dselect/dpkg

14.1.2 Obtendo as imagens do disquete

Se você tem uma conexão rápida com a Internet, a melhor maneira de conseguir o Debian é por FTP anônimo. O servidor de ftp para o Debian está em ftp.debian.org , e o diretório é /debian/dists . A estrutura do pacote é descrita na tabela a seguir:

Diretório	Conteúdo
./stable/main	Última versão estável

./stable/main/binary-i386	Pacotes Debian para arquitetura Intel i386
./stable/main/disks-i386	Discos de inicialização e root necessários para a instalação
./stable/main/disks-i386/current	Os discos atuais

Tabela 2.1: Estrutura do pacote do Debian GNU/Linux

Para uma instalação básica do Debian, você vai precisar apenas de 12 MB de espaço em disco e alguns disquetes. Primeiro, você precisa gerar os disquetes de inicialização. No diretório /debian/dists/stable/main/disks-i386/current/images-1.44 (para disquetes de 1.44MB), você vai encontrar:

Nome do arquivo	Rótulo	Descriç o
rescue.bin	Disquete de resgate	
driver- (1 até 4).bin	Drivers de dispositivos	
base-(1 até 11).bin	Disquetes base	

Tabela 2.2: Disquetes de instalação para Debian GNU/Linux.

Você também pode usar o conjunto fornecido no diretório compact, que seguramente servirá para iniciar a instalação.

Escolha o conjunto apropriado de disquetes para o seu hardware a partir da tabela acima e gere-os como descrevemos anteriormente.

14.1.3 Fazendo o Download dos pacotes

Para instalar e usar o Debian, você vai precisar mais que o sistema base. Para decidir quais pacotes você precisa, baixe o arquivo Packages de:

ftp://ftp.debian.org/debian/dists/stable/main/binary-i386/Packages

Este arquivo é a listagem corrente dos pacotes Debian disponíveis na versão estável. O arquivo tem um formato especial: cada pacote tem uma entrada e está separado por uma linha em branco. A informação de cada pacote está separada em campos. A tabela 2.3 descreve os campos e os possíveis valores. Ele deve dar uma idéia de como montar sua lista pessoal. Quando tiver a listagem de pacotes que quiser, você terá de decidir como fazer o download dos mesmos. Se for um usuário experiente, pode querer fazer o download do pacote netbase e assim poderá fazer o download dos pacotes posteriormente, já no Linux. De outra forma, você pode fazer o download de todos os pacotes com o seu sistema operacional atual e instalá-los posteriormente a partir de uma partição montada.

Campo	Descrição	
Package	Nome do pacote	
Priority	Importância do pacote Required -	

Section	necessário para o funcionamento do sistema Important - não necessário, mas importante Optional - não necessário, mas importante Optional - não necessário, mas útil Extra - pode conflitar com outros pacotes de mais alta prioridade Categoria Geral Base Sistema base Devel Ferramenta de desenvolvimento X11 pacote para o X11 Admin ferramentas para administraç o Doc documentaç o Comm pacotes para comunicaç o Editors editor de texto Electronics ferramentas para eletrônica Games jogos Graphics ferramentas gráficas Hamradio ferramentas para rádio para Internet Mail servidores e clientes de e-mail Math ferramentas matemáticas Net ferramentas para conex o rede News servidores e clientes de NNTP Shells shells, como o bash, o sh, etc Sound aplicativos de som TeX aplicativos para o TeX Text ferramentas para manipular textos Misc qualquer coisa fora das categorias	
Installed-Size	acima Tamanho total instalado	
Maintainer	Quem mantém o pacote	
Architecture	Arquitetura para a qual o pacote foi escrito	
Version	Versão do pacote	
Depends	Dependência de outros pacotes,se houver.	
Filename	Arquivo onde está o pacote	
Size	Tamanho do arquivo do pacote	
MD5Sum	Checksum para verificaç o de integridade do pacote	
Description	Breve descriç o do pacote	

 Tabela 2.3: Campos do arquivo Packages do Debian/GNU Linux.

14.1.4 Inicializando dos disquetes e instalando o Debian GNU/Linux

14.1.4.1 O disco de rescue

Insira o disquete Rescue e reinicialize. Em alguns minutos, você verá a tela de introdução do Rescue e o prompt de inicialização.

Ele é chamado de disquete Rescue (resgate, salvamento) porque pode ser usado para inicializar o sistema e reparar algum problema que torne seu disco não inicializável. Mantenha esse disquete mesmo após instalar o sistema.

Você pode fazer duas coisas no prompt boot: : pressione as teclas F1 até F10 para ver algumas telas com informações úteis, ou inicialize o sistema. Se você tem qualquer dispositivo de hardware que o Linux não acesse corretamente durante a inicialização, pode encontrar um parâmetro para adicionar ao comando de inicialização do sistema nas telas de ajuda ao pressionar F3, F4 e F5.

Se adicionar parâmetros à linha de comando da inicialização, esteja certo de digitar a palavra "linux" e um espaço antes do primeiro parâmetro. Se simplesmente pressionar <Enter>, isso será o mesmo que ter digitado "linux" sem parâmetro algum.

Se esta é a primeira vez que você está inicializando o sistema, pressione <Enter> para ver se funciona corretamente. Provavelmente irá funcionar. Se não, você pode reinicializar mais tarde e verificar se precisa de algum parâmetro especial que informe ao sistema sobre seu hardware.

Uma vez que tenha pressionado <Enter>, deverá ver as seguintes mensagens:

Loading...

Uncompressing Linux...

Depois virá uma página ou quase isso com informação codificada sobre seu hardware. Poderão aparecer várias mensagens do tipo "can't find" (não posso encontrar) alguma coisa", "alguma coisa not present (não está presente)", "can't initialize (não posso inicializar)qualquer coisa", ou ainda "this driver release depends on (esta versão do driver depende de) qualquer coisa". A maioria destas mensagens não faz mal algum. Os disquetes de inicialização são preparados para rodar com muitos computadores diferentes e diferentes dispositivos. Obviamente, nenhum computador terá todos os dispositivos possíveis e o sistema operacional vai reclamar cada vez que procurar por um dispositivo não presente. Você poderá notar que algumas vezes ele dará uma pausa. Isso acontece porque o sistema estará aguardando pela resposta do dispositivo que pode não estar presente. Se você achar que a inicialização do sistema está inaceitavelmente longa, pode criar um kernel personalizado após instalar o sistema, que não tenha todos os drivers dos dispositivos não existentes.

14.1.4.2 A caixa de diálogos colorido ou monocromático

Uma vez que tenha terminado de inicializar, você deve ver a caixa de diálogo para selecionar colorido ou monocromático. Selecione preferencialmente "color", a não ser que você tenha um monitor monocromático.

14.1.4.3 O Menu Principal

Você poderá ver uma caixa de diálogos que diga:

The installation program is determining the current stat of your system. (O programa de instalação está determinando o estado atual do seu sistema)

Em alguns sistemas, esta mensagem pisca muito rápido para ser lida. Ela é mostrada entre os passos do processo de instalação. O programa de instalação verifica o estado do sistema após cada passo. Isso permite que você reinicie a instalação sem perder o trabalho que já fez, se por acaso parar o sistema no meio da instalação. Se precisar reiniciar uma instalação, você será questionado sobre o tipo de monitor novamente (colorido ou monocromático), configuração do teclado, reativar partição de swap e remontar qualquer disco que tenha sido inicializado. Qualquer outra instalação no sistema terá sido salva.

Durante todo o processo você poderá acessar o menu principal. As opções no topo do menu mudam para indicar seu progresso na instalação do sistema. Phil Hughes escreveu no *Linux Journal* que poderia ensinar um frango a instalar o Debian. Ele quis dizer que o processo de instalação é principalmente "bicando" a tecla <Enter>. A primeira opção do menu é a próxima ação que você deve tomar de acordo com o que o sistema detectou que você já fez. Ela deve ser Next (próximo) e, neste ponto, a próxima coisa deve ser a configuração do teclado.

14.1.4.4 Configuração do teclado

Esteja certo que está sobre o item Next e pressione <Enter> para entrar no menu de configuração do teclado. Selecione um teclado que esteja adequado ao padrão do seu país ou algum com configuração próxima se o seu não é mostrado. Depois da instalação você poderá selecionar um teclado de um conjunto maior de opções. Mova a barra para o teclado escolhido e pressione <Enter>.

14.1.4.5 O shell

Se você é um usuário experimentado UNIX ou Linux, pressione Alt da esquerda mais F2 para um segundo console virtual. Você verá uma segunda janela rodando um clone do Bourne shell chamado ash. Neste ponto, o sistema de arquivos root está em um disco virtual na memória RAM e há poucas ferramentas UNIX disponíveis para uso. Você pode ver os programas disponíveis através do comando:

Is /bin /sbin /usr/sbin

O shell e os comandos estão aí somente em caso de algo dar errado. Em particular, você deve usar sempre os menus, não o shell, para ativar a partição de swap, porque o programa do menu não pode determinar se você fez isso através do shell. Pressione Alt da esquerda mais F1 para voltar ao menu. O Linux suporta até 64 consoles virtuais, mas o disco de resgate usa poucos deles.

Última Chance! Já fez backup de seus discos? Aqui está sua primeira chance de apagar todos os dados em seus discos, e sua última chance de salvar seu sistema antigo. Se você não fez backup, remova o disquete do drive, reinicie o sistema e faça-o.

14.1.4.6 Particionando seus discos

Se você ainda não particionou seus discos para os sistemas de arquivo Linux nativo e Linux Swap, o menu Next vai ser "Partition a Hard Disk" (Particionando o disco)

Se você já criou ao menos uma partição Linux nativa e uma de swap, a opção Next será "Initialize and Activate the Swap Disk Partition" (Inicializar e Ativar a Partição de Swap), ou pode até pular esta etapa se seu sistema tem pouca memória RAM e o programa de instalação perguntou se você queria ativar a partição de swap assim que o sistema iniciou. Qualquer que seja o caso, você ainda pode usar as setas para selecionar a opção "Partition a Hard Disk"

O menu "Partition a Hard Disk" apresenta uma lista de discos que você pode particionar e roda o cfdisk (já comentado anteriormente), que permite criar e editar as partições no disco. Você deve criar ao menos uma partição Linux (tipo 83).

Sua partição de swap será usada para prover memória virtual ao sistema e deve ter entre 16 e 128MB de tamanho, dependendo de quanto espaço em disco você e tenha e quão grande sejam os programas que queira rodar. O Linux não vai usar mais de 128 MB de swap, por isso não há motivo para criar uma partição de swap maior. Uma partição de swap é fortemente recomendada, mas você até pode rodar sem ela.

14.1.4.7 Inicializando e Ativando a Partição de Swap

"Initialize and Activate the Swap Disk Partition" é o próximo menu ao pressionar Next após criar as partições. Você tem a opção de inicializar e ativar uma partição de swap nova, ativar uma partição já inicializada, ou continuar sem ativar uma partição de swap. A menos que você saiba o que está fazendo, opte por "Initialize and Activate the Swap Disk Partition" (inicializar e ativar uma partição de swap). Esta opção lhe permitirá verificar a partição e identificar possíveis setores defeituosos. Isto é útil se você tem um disco MFM, RLL ou antigos IDE, e verificar o disco nunca é demais. Discos SCSI usados adequadamente não precisam ser verificados. Eles têm seu próprio mecanismo de mapear setores defeituosos.

A partição de swap fornece memória virtual para complementar a memória RAM no seu sistema e é usada durante a instalação. Por isso é a primeira coisa a fazer.

14.1.4.8 Inicializando uma partição Linux

Neste ponto, o próximo menu deve ser "Initialize a Linux Partition" (inicializando uma partição Linux). Se não for, você não deve ter completado o processo de particionamento do disco, ou não ter selecionado uma das opções sobre a particão de swap.

Você pode inicializar uma partição do Linux ou montar uma partição já inicializada anteriormente.

Os disquetes de inicialização não vão fazer um upgrade do sistema sem

remover os arquivos - o Debian tem um procedimento diferente para fazer upgrade. Assim, se você estiver usando partições antigas que não estão vazias, você deve inicializá-las, o que apaga qualquer arquivo. Você deve inicializar qualquer partição criada no passo de particionamento do disco. A única razão de montar uma partição sem inicializá-la neste ponto seria a montagem de partições com arquivos de usuários, como o /home, que você não vai querer apagar.

Selecione a opção Next no menu para inicializar e montar a partição root (o diretório "/"). A primeira partição que você montar, depois da partição de swap (se estiver usando uma), é a partição root. Você poderá escolher por verificar o disco contra setores defeituosos, assim como quando inicializou a partição de swap. Não é uma má idéia, mas mantenha em mente que este passo pode levar mais de 10 minutos, dependendo do tamanho de seu disco.

14.1.4.9 Instalando o sistema base

Após ter montado a partição root, o próximo menu será "Install the Base System" (instalando o sistema base), a não ser que você já tenha feito alguns passos da instalação. Se você tem partições adicionais para configurar, use as setas para selecionar as opções de inicialização e montagem dos discos. Se você criou partições separadas para /var, /usr, ou outros sistemas de arquivos, deve inicializá-las e montá-las agora.

Haverá uma pausa enquanto o sistema procura por uma cópia local do sistema base. Esta procura é para instalações por CD-ROM e não terá êxito. Você verá, então, um menu com drives de onde ler os disquetes base. Selecione o drive apropriado. Vá colocando os disquetes Base 1, Base 2 e assim por diante, conforme requisitado pelo programa. Se um dos disquetes tiver erro de leitura, você terá de criar um novo e colocar todos os discos no sistema novamente. Após ler todos os disquetes, o sistema instala os arquivos. Isso pode levar vários minutos.

14.1.4.10 Instalando o kernel do sistema operacional

Neste ponto, o próximo menu deve ser "Install the Operation System Kernel" (instalando o kernel do sistema operacional). O programa solicitará que selecione um drive de disquete para inserir o disco de resgate. Este processo copia o kernel no disco rígido. Este kernel é usado posteriormente para criar um disquete de inicialização para seu sistema e fazer seu disco rígido inicializável sem disquete.

14.1.4.11 Instalando os drivers de dispositivos

Selecione o item de menu para instalar os drivers de dispositivos. O programa solicitará que insira os disquetes de Drivers e os mesmos serão copiados para seu disco rígido. Selecione o item "Configure Device Drivers" e procure dispositivos que estejam em seu sistema. Configure esses dispositivos para que eles sejam carregados sempre que você reinicie o sistema.

Há uma opção de menu para drivers PCMCIA, mas você não precisa usá-la. Depois da instalação, você poderá instalar o pacote pacot

detecta as placas PCMCIA automaticamente e configura aquelas que encontrar. Ele também reconhece placas "hot swap" enquanto o sistema estiver rodando - elas serão configuradas quando conectadas e desconfiguradas quando desconectadas.

14.1.4.12 Configurando o sistema base

Neste ponto o sistema lê toda a informação que faz parte de um sistema Debian mínimo, mas você terá que fazer algumas configurações antes de rodar o sistema. Selecione "Configure the Base System" (configurar o sistema base). Isto irá pedir que você selecione a time zone (fuso horário). Verifique seu fuso horário (grande parte do Brasil usa o fuso Brasília - GMT -03:00). Os sistemas UNIX usam a hora GMT no relógio do sistema e um software que converte isso para o horário local). Isso permite que ele possa lidar com horário de verão e anos bissextos, e até que usuários remotos entrem no sistema e configurem seu fuso horário individualmente para o seu terminal. Se você roda o sistema com GMT e sua localidade tem horário de verão, o sistema ajusta automaticamente o dia de início e fim do horário de verão.

14.1.4.13 Configurando a rede

Você deve configurar a rede, mesmo que você não tenha uma, mas neste caso terá que responder somente às duas primeiras perguntas:

What is the name of your computer? (Qual o nome de seu computador)

е

Is your system connect to a network? (Seu sistema está conectado a uma rede)

Se você estiver conectado a uma rede, verifique com o administrador da mesma ou com o provedor de acesso se não tiver as seguintes informações:

- o nome de seu computador;
- o domínio onde ele está (pode ser da sua rede ou do provedor);
- o endereço IP do seu computador;
- a máscara de rede:
- o endereço IP inicial da sua rede;
- o endereço de broadcast;
- se a rede tiver um gateway, o endereço IP do gateway para onde você deve rotear os pacotes;
- o endereço do seu servidor de DNS (Domain Name Service); e
- se você conecta na rede usando Ethernet.

O programa irá deduzir que o endereço IP de sua rede é um E lógico (AND) do seu endereço IP com a sua máscara. Também irá deduzir que o endereço de broadcast é o OU lógico (OR) do seu endereço IP com a negação da máscara. Irá também deduzir que o seu gateway é o mesmo que seu servidor DNS. Se você não tiver estar respostas, use as deduções do sistema - se necessário, poderá alterá-las posteriormente editando o arquivo

14.1.4.14 Tornando seu disco rígido inicializável

Se você escolheu que seu disco inicialize direto no Linux, o programa irá perguntar se você quer instalar um MBR (master boot record). Se você não estiver usando o gerenciador de inicialização (provavelmente este é o caso se você não souber o que é isso), responda "yes" à pergunta. A próxima pergunta é se você quer inicializar o Linux automaticamente a partir do disco rígido quando ligar o sistema. Isso faz com que a partição do Linux seja inicializável - aquela que será carregada do disco rígido quando o sistema for ligado. Se você responder "no", poderá configurar a partição como inicializável posteriormente com o FDISK.EXE do windows ou o fdisk do Linux.

14.1.4.15 Criando um disquete de inicialização

Você deve criar um disquete de inicialização, mesmo que queira iniciar seu sistema pelo disco rígido. A razão para isto é que é possível que o disco seja gravado incorretamente. O disquete provavelmente irá funcionar nesse caso. Selecione "Make a Boot Floppy" (Fazer um disquete de Inicialização) pelo menu e insira um disquete virgem desprotegido no sistema quando indicado. O programa vai tentar formatá-lo e gravar nele. Marque o disquete como "Custom Boot" e proteja-o assim que o processo terminar.

14.1.4.16 O momento da verdade

Esta é a parte que os engenheiros elétricos chamam de "teste de fumaça" - o que acontece quando você liga um sistema pela primeira vez. Remova o disquete do drive e selecione "Reboot the System" (Reiniciar o sistema) a partir do menu. Você deverá ver as mesmas mensagens de quando inicializou durante a instalação, seguidas de algumas novas mensagens.

14.1.4.17 Add a user account and password

Depois de ter adicionado logins, você é jogado no deselect, o programa de gerenciamento de pacotes do Debian.

Você deve ler o tutorial antes de tentar instalar qualquer pacote com o dselect.

dselect permite que você selecione pacotes que queira instalar no sistema. Veja a seção sobre dselect - gerenciador de pacotes do Debian - mais adiante. Se você tem um CD-ROM ou disco rígido com os pacotes adicionais do Debian ou está conectado na Internet, você pode querer ler esse seção agora. Senão, saia do dselect. Você pode usar o gerenciador de pacotes depois de ter transferido os arquivos com os pacotes do Debian para seu sistema.

Você deve ser um super-usuário (root) para usar o dselect.

Se você instala o X Window e não usa um teclado norte-americano (US), leia a nota do X11 sobre "non-US keyboards" (teclados não-americanos).

14.1.4.18 Log in

Depois de sair do dselect, você está no prompt do login:. Entre usando seu login pessoal e a senha que você escolheu. Você está pronto para usar o sistema.

14.1.5 Rodando o Debian GNU/Linux

Esta seção descreve o sistema de pacotes Debian e as ferramentas específicas do Debian. O formato do arquivo Packages é mostrado na tabela 2.3.

A distribuição Debian vem em arquivos chamados **pacotes**. Cada pacote é uma coleção de arquivos (usualmente programas) que podem ser instalados usando o dpkg ou dselect. Além disso, o pacote contém informação sobre si mesmo que é lida pelas ferramentas de instalação.

14.1.5.1 Classificação dos Pacotes

Os pacotes que foram incluídos no Debian GNU/Linux são classificados de acordo com sua necessidade (item priority - prioridade) e sua funcionalidade (item section - seção)

A prioridade do sistema indica quão necessário ele é. O Debian GNU/Linux classifica todos os pacotes em quatro diferentes níveis de prioridade:

14.1.5.1.1 Required (Necessário)

Estes pacotes devem ser instalados para que o sistema funcione corretamente e são instalados como parte do sistema base.

Nunca remova um pacote necessário do sistema a não ser que você esteja absolutamente certo do que está fazendo. Isto é bom repetir: nunca, nunca, nunca remova um pacote necessário do sistema a não ser que você esteja absolutamente certo do que está fazendo. Fazendo isso, provavelmente seu sistema ficará completamente inutilizado.

Pacotes Required são abreviados no dselect como Req.

14.1.5.1.2 Important (Inportante)

Pacotes importantes são encontrados em quase todos os sistemas operacionais tipo UNIX. Estes pacotes incluem cron, man, e vi.

Pacotes Importantes são abreviados como Imp.

14.1.5.1.3 Standard (Padrão)

Pacotes Standard são pacotes que compreendem, mais ou menos, o sistema padrão do Debian GNU/Linux. O sistema padrão inclui um pacote bem completo de ferramentas de desenvolvimento e o GNU Emacs.

Pacotes Standard dselect como Std.

14.1.5.1.4 Optional (Opcional)

Pacotes opcionais compreendem vários sistemas. Dentre eles, o X Window e o Tex.

São abreviados como opt.

14.1.5.1.5 Extra (Extra)

Pacotes Extras são úteis somente para um pequeno grupo de pessoas, ou instalados para um propósito específico. Podem incluir programas de eletrônica ou aplicações de rádio.

Pacotes Extras são abreviados como Xtr.

Por default, dselect seleciona automaticamente os pacotes Standard se o usuário não quiser selecionar individualmente os pacotes a serem instalados.

A seção (section) de um pacote indica sua funcionalidade ou uso. Pacotes em CD-ROM ou em repositórios FTP são organizados em subdiretórios de acordo com sua função. Os nomes dos diretórios são bem auto-explicativos: o diretório admin, por exemplo, contém pacotes para administração do sistema e o diretório devel contém pacotes de software para desenvolvimento e programação. Diferentemente dos níveis de prioridade, há diversas "sections", e outras podem ser adicionadas no futuro, assim, não as descreveremos individualmente neste guia.

14.1.5.2 Relacionamento entre Pacotes

Cada pacote inclui informação sobre como ele se relaciona com outros pacotes incluídos no sistema. Há quatro relacionamentos de pacotes no Debian: conflito, dependência, recomendação e sugestão (conflicts, dependencies, recommendations, and suggestions).

Um conflito ocorre quando dois ou mais pacotes não podem ser instalados no mesmo sistema ao mesmo tempo. Um bom exemplo de pacotes conflitantes são os agentes de transferência de mail (MTAs). Um MTA é um programa que envia e-mails para usuários no sistema e em outras máquinas na rede. o Debian suporta dois agentes de e-mail: sendmail ou smail.

Somente um agente de transferência de e-mail pode ser instalado de cada vez. Eles fazem a mesma função e não são projetados para coexistir. Assim, os pacotes sendmail e smail conflitam. Se você tentar instalar o sendmail quando o smail já estiver instalado, o sistema de gerenciamento de pacotes do Debian vai se recusar a instalá-lo. Da mesma forma, se tentar instalar o smail quando o sendmail já estiver instalado, o dselect (ou dpkg; veja abaixo) se recusará a instalá-lo.

Uma dependência ocorre quando um pacote requer outro pacote para funcionar corretamente. Usando o exemplo do e-mail, usuários lêem suas mensagens com programas chamados agentes de e-mail (MUAs - mail user agents). MUAs populares incluem elm, pine e Emacs. É normal instalar vários MUAs simultaneamente porque eles não são conflitantes. Mas MUAs não enviam e-mail - esse é um trabalho dos MTAs. Assim, todos os MUAs dependem de um MTA.

Um pacote também pode recomendar ou sugerir outros pacotes relacionados.

14.1.6 dselect

Esta seção é um breve tutorial sobre o deselect do Debian. Para

informações mais detalhadas, refira-se ao manual do deselect em ftp://ftp.debian.org/debian/dists/stable/main/disks-i386/current/doc/dselect-beginner.en.html

dselect é uma interface simples, orientada a menus, que ajuda a instalar os pacotes. Ele leva você através do processo de instalação dos pacotes na ordem das opções do menu:

Debian Linux dselect package handling front end.

- 0. [A]ccess Choose the access method to use.
- 1. [U]pdate Update list of available packages, if possible.
- 2. [S]elect Request which packages you want on your system.
- 3. [I]nstall Install and upgrade wanted packages.
- 4. [C]onfig Configure any packages that are unconfigured.
- 5. [R]emove Remove unwanted software.
- 6. [Q]uit Quit dselect.

Há duas maneiras de selecionar a opção do menu: escolha a opção usando as setas, ou pressione a tecla correspondente entre colchetes.

14.1.6.1 Access (Acesso)

Neste item você escolhe o método de acesso para obter e instalar os pacotes.

Abbreviation	Description
cdrom	install from a CD-ROM
nfs	install from an NFS server (not yet mounted)
harddisk	install from a hard disk partition (not yet mounted)
mounted	install from a file system which is already mounted
floppy	install from a pile of floppy disks
ftp	install using ftp

14.1.6.2 Update (Atualização)

dselect lê a base de dados de pacotes e cria uma base de dados dos pacotes disponíveis para o seu sistema.

14.1.6.3 Select (Selecionar)

Este item seleciona os pacotes. Escolha os pacotes que você quer e pressione <Enter>. Se você tem uma máquina lenta, a tela pode ficar branca por uns 15 segundos. A primeira coisa que aparece é a primeira página do arquivo de Ajuda. Você pode ver esta tela pressionado "?" em qualquer momento que estiver na janela de seleção, e pode passar as páginas pressionando a tecla ponto (.).

Para sair da tela de seleção, após todas as seleções completas, pressione <Enter>. Você retornará à tela principal se não houver problemas com sua seleção. Se houver, você terá de resolvê-los primeiro. Quando estiver satisfeito com qualquer tela, pressione <Enter>

Conflitos de dependência são bem normais e até esperados. Se você selecionou um pacote A e esse pacote requer o pacote não selecionado B para rodar, o dselect avisa sobre o problema e normalmente sugere a solução. Se o pacote A conflita com o pacote B, você terá de optar entre eles.

14.1.6.4 Install (Instalar)

dselect passa pelos 800 pacotes e instala aqueles que você selecionou. Você terá de tomar algumas decisões sobre o processo. Usualmente é útil abrir um outro shell para comparar, por exemplo, uma configuração antiga com a nova. Se a antiga é chamada conf.modules, por exemplo, a nova será conf.modules.dpkg-new.

A tela rola bem rápido em máquinas velozes. Você pode parar a rolagem pressionando Control-S e reiniciar com Control-Q. No final, sobrará uma lista de pacotes não instalados.

14.1.6.5 Configure (Configurar)

A maior parte dos pacotes é configurada no passo 3, mas o que sobrar pode ser configurado aqui.

14.1.6.6 Remove (Remover)

Remover os pacotes que não são mais necessários.

14.1.6.7 Quit

Au revoir.

14.1.7 dpkg

Esta é uma ferramenta por linha de comando que instala que instala e manipula pacotes Debian. Ela tem muitas opções que permitem instalar, configurar, remover ou realizar outras operações nos pacotes Debian. Você pode até criar seus próprios pacotes. dpkg também permite listar todos os pacotes disponíveis, arquivos pertencentes aos pacotes, que pacotes detêm um arquivo, e assim por diante.

14.1.7.1 Instalando ou atualizando pacotes novos ou existentes

Digite o seguinte comando:

dpkg -i arquivo.deb

Onde arquivo é o nome do arquivo que contém o pacote Debian (ex: tcsh_6.06-11_i386.deb). O dpkg é parcialmente interativo. Durante a

instalação ele pode fazer perguntas adicionais, assim como instalar ou não uma nova versão de um arquivo de configuração ou manter a versão antiga.

Você também pode desempacotar um pacote sem configurá-lo, entrando:

dpkg --unpack arquivo

Se um pacote depende de outro não instalado ou de uma nova versão de um que você já tenha, ou se qualquer outro problema de dependência ocorrer durante a instalação, dpkg sairá sem configurá-lo.

14.1.7.2 Configurando pacotes instalados

Se o dpkg aborta durante a instalação e deixa um pacote instalado, o pacote não foi configurado. O sistema de pacotes do Debian precisa que o pacote seja configurado para evitar problemas de dependência. Alguns pacotes também necessitam ser configurados para funcionar corretamente.

Para configurar um pacote, digite:

dpkg --configure pacote

Onde pacote é o nome do pacote, como tcsh (note que este não é o nome original do arquivo de onde o tcsh foi instalado, que era maior, incluindo o número da versão e terminado em .deb).

14.1.7.3 Removendo pacotes instalados

No sistema de pacotes do Debian, há duas maneiras de eliminar pacotes: as opções remove e purge (remover e limpar). A opção remove remove o pacote especificado; a opção purge remove tanto o pacote especificado quanto seus arquivos de configuração. O uso é:

dpkg -r package

ou

dpkg --purge package

Se houver pacotes instalados que dependam daquele que você quer remover, o pacote não será removido, e o dpkg irá abortar com uma mensagem de erro.

14.1.7.4 Relatando a situação dos pacotes

Para informar a situação dos pacotes (ex: instalado, não instalado, ou não configurado), digite:

dpkg -s package

14.1.7.5 Listando pacotes disponívies

Para listar os pacotes disponíveis que fechem com um padrão, digite:

dpkg -l padrão-de-nome-de-arquivo

Onde padrão-de-nome-de-arquivo é um argumento opcional

especificando o padrão para os nomes dos pacotes a encontrar, como *sh. Caracteres coringa do shell são permitidos. Se você não especifica um padrão, todos os pacotes instalados são listados.

14.1.7.6 Listando arquivos pertencentes por pacotes

Para listar todos os arquivos pertencentes a um pacote em particular, simplesmente digite:

dpkg -L pacote

No entanto, isso não lista os arquivos criados por scripts de instalação específicos dos pacotes.

14.1.7.7 Encontrando o pacote que detem um arquivo

Para encontrar o pacote ao qual o arquivo pertence, digite o seguinte comando:

dpkg -S padrão-de-nome-de-arquivo

Onde *padrão-de-nome-de-arquivo* é o padrão para procurar nos nomes de pacotes. Novamente, caracteres coringa do shell são permitidos.

14.1.7.8 Sumário

dpkg é simples de usar e é preferido sobre o dselect quando tudo que você tiver de fazer é uma instalação, atualização ou remoção de um pequeno número de pacotes. Ele tem também alguma funcionalidade que dselect (que é uma interface para o dpkg) não tem, como encontrar o pacote ao qual pertence um arquivo. Para uma lista exaustiva de opções, refira-se à página do manual do dpkg (8).

14.1.8 Sobre o Debian GNU/Linux

O Projeto Debian GNU/Linux foi criado por lan Murdock em 1993, inicialmente sob o patrocínio do projeto GNU da Free Software Foundation (FSF). Posteriormente, o Debian se separou da FSF. O Debian é o resultado de um esforço voluntário para criar um sistema operacional livre, de alta qualidade, compatível com o UNIX e baseado no kernel do Linux, complementado com um conjunto de aplicações.

A comunidade Debian é um grupo com mais de 150 voluntários não remunerados de todo o mundo, que colaboram via Internet. Os fundadores do projeto formaram a organização *Software in the Public Interest* (SPI) para patrocinar o desenvolvimento do Debian GNU/Linux.

Software in the Public Interest é uma organização sem fins lucrativos formada quando a FSF retirou o patrocínio ao projeto Debian. A finalidade da organização é desenvolver e distribuir software livre. Seus objetivos são muito similares aos da FSF, e ela encoraja programadores a usar a licença GNU nos seus programas. No entanto, a SPI tem um foco levemente diferente ao construir e distribuir um sistema Linux que diverge em muitos detalhes técnicos do sistema GNU planejado pela FSG. SPI ainda se comunica com a FSF e

coopera enviando alterações ao software GNU e solicitando que seus usuários doem recursos à FSF e ao projeto GNU.

A SPI pode ser encontrada em:

http://www.spi-inc.org/

ou

Software in the Public Interest P.O. Box 70152 Pt. Richmond, CA 94807-0152

14.2 LINUX RED HAT

Instalação do Red Hat

Esta seção foi escrita por Henry Pierce.

14.2.1 Propriedades da instalação do Linux Red Hat Linux

Dependências: sim

Método de inicialização da instalação: CD, disquete,

loadin (a partir do DOS)

Métodos de instalação: CD, HD, NFS, FTP,

SMB

Inicialização do sistema: Sys V init Facilidade de instalação: fácil Ferramentas gráficas de administração: várias

Ferramenta de instalação: script de instalação

chamado RPM

Ferramenta para manutenção de pacotes: RPM, glint

14.2.2 O sistema de gerenciamento de pacotes RPM

O sistema de gerenciamento de pacotes do Red Hat (RPM) gerencia o software recolhendo informação sobre componentes e métodos de instalação do pacote durante o processo de construção. Um pacote RPM tem um cabeçalho com dados organizados que podem ser adicionados a uma base de dados que descreve ao que esse pacote pertence, quais outros pacotes são necessários, se os pacotes necessários estão instalados, e fornece uma maneira de determinar dependências de pacotes.

O RPM dá aos administradores do sistema a habilidade de: atualizar componentes individuais ou sistemas inteiros preservando a configuração do sistema ou pacote; pesquisar a base de dados sobre a localização dos arquivos, pacotes, ou outra informação relacionada; realizar verificações nos pacotes para instalá-los corretamente; manter os fontes "puros" (manter as fontes do pacote do autor original e remendos - patches - de outros autores separadamente), para que problemas de portabilidade possam ser rastreados.

Por esse motivo, você instalar, atualizar ou remover pacotes com uma simples linha de comando no modo texto ou poucos cliques de mouse no X. Exemplos usando o rpm por linha de comando são:

rpm --install pacote.rpm # isto instala pacote

rpm --upgrade pacote.rpm # isto atualiza pacote

rpm --erase pacote.rpm # isto remove ou apaga pacote

14.2.2.1 Convenções de nome dos pacotes

Um pacote RPM bem construído tem as seguintes características: o nome identifica o pacote, a versão, o número da revisão, a arquitetura; e a extensão .rpm o identifica como um pacote RPM.

Peguemos, por exemplo, bash-1.14.7-1.i386.rpm. O nome por si só já contém informação útil: o pacote é o bash (o Bourne Again SHell), sua versão é 1.14.7, e sua revisão é 1 em relação à versão corrente para o Red Hat. Foi construído para o Intel 80386 ou compatível e está no formato RPM. Assim, se você vir um pacote chamado bash-1.14.7-2.i386.rpm, saberá que é a segunda revisão do bash versão 1.14.7, e provavelmente contém correções para versões anteriores e é mais atual. A organização interna de um arquivo rpm está além do escopo desta seção, mas um pacote bem construído contém um arquivo executável, os arquivos de configuração, a documentação (no mínimo as páginas do manual), arquivos variados relacionados ao pacote, um registro indicando onde os arquivos devem ser instalados e um registro indicando os pacotes que são pré-requisitos. Depois da instalação bem sucedida, uma informação sobre o pacote é registrada na base de dados RPM do sistema. Uma discussão mais completa do sistema de gerenciamento de pacotes RPM pode ser encontrada em http://www.rpm.org.

14.2.3 Uma nota sobre atualização do Red Hat

Atualizações do Red Hat somente são suportadas a partir da versão 2.0 devido a grandes mudanças no formato binário do Linux. Atualizações podem ser feitas através dos mesmos métodos de instalação (CD-ROM, NFS, FTP e disco rígido). A partir da versão 4.0, a versão de atualização também está disponível a partir do disquete de atualização, ao invés de um programa separado.

14.2.4 Criando disquetes de instalação

Para criar um kit de disquetes de inicialização, você precisa:

- 1. da imagem do disquete de inicialização Red Hat, que está disponível em ftp://ftp.redhat.com/pub/redhat/linux/current/en/os/i386/images/boot.img, ou no diretório images do CD-ROM do Red Hat.
- 2. dos disquetes de drivers suplementares, isto se você não for instalar a partir de um CD-ROM

drvblock.img - Supplemental Block Device Drivers drvnet.img - Supplemental Network Drivers

Eles estão no mesmo diretório do boot.img.

- 3. do programa RAWRITE.EXE disponível em ftp://ftp.redhat.com/pub/redhat/linux/current/en/os/i386/dosutils/rawrite.exe, ou no CD-ROM do Red Hat
- 4. do FIPS.EXE usuários Windows instalando Red Hat pela primeira vez na máquina, podem ter que obter o FIPS.EXE do endereço ftp://ftp.redhat.com/pub/redhat/linux/current/en/os/i386/dosutils/fips.exe para liberar espaço na sua máquina.
- 5. criar um disquete de inicialização para o sistema operacional que está rodando atualmente na máquina. Este item é para sua segurança.

14.2.5 Mídia de instalação

Depois de criar os disquetes de instalação usando o RAWRITE.EXE, assegure-se que seu método de instalação está bem configurado para os disquetes do Red Hat. Para CD, NFS, FTP e discos rígidos, a fonte deve conter o diretório /RedHat no nível raiz, com os diretórios /base e /RPMS abaixo:

/RedHat

|-----> /RPMS (contem os pacotes a serem instalados)

|-----> /base (contem o sistema base e arquivos para configurar o disco rígido)

14.2.5.1 Instalação por NFS

Para instalação por NFS, você ou vai precisar de um CD-ROM Red Hat numa máquina que suporte e exporte sistema de arquivos ISO-9660 (sistema de arquivos padrão do CD) com Extensões Rockridge, ou vai precisar de uma cópia da distribuição Red Hat num diretório organizado como descrito acima e, obviamente, os arquivos dentro de cada diretório. O diretório /RedHat precisa ser exportado para as máquinas da rede que terão o Red Hat instalado ou atualizado. Essa máquina precisa estar na Ethernet. Você não pode fazer uma instalação NFS por link discado.

14.2.5.2 Instalação por disco rígido

Instalações por disco rígido precisam do diretório /RedHat criado na raiz da partição (não importando qual partição) contendo a distribuição obtida de um CD ou site de FTP. Num sistema de arquivos MS-DOS, não importa que os nomes dos pacotes.rpm estejam truncados. Você precisa apenas se certificar que o diretório \RedHat\base contenha os arquivos base copiados de um CD ou site de FTP e que \RedHat\RPMS contenha todos os arquivos pacotes.rpm do CD ou site de FTP. A partir daí você poderá instalar daquela partição.

Instalação por FTP

Para instalar via FTP pela Internet, você precisa do endereço IP do servidor de FTP e do diretório raiz para o sistema Red Hat que quer instalar. Se você quer fazer uma instalação FTP com pouca largura de banda (qualquer

coisa menos que 128Kbps), é melhor copiar os arquivos primeiramente para o disco rígido. O tamanho total dos pacotes no diretório RPMS é de aproximadamente 1GB e vai levar várias horas para instalar. Se algo der errado com a instalação (o link cair, por exemplo), você precisa começar tudo novamente. Se você pegar os arquivos antes, será menos trabalhoso e confuso para reiniciar de uma instalação que tenha falhado. Você nem precisa fazer o download de todos os arquivos do diretório RPMS para uma instalação mínima do Red Hat. Veja a próxima seção para detalhes.

14.2.6 Personalizando sua instalação por disco rígido ou NFS

Você pode personalizar o processo de instalação. Somente os mais familiarizados com o Linux deveriam tentar. Na versão 7.2 do Red Hat o diretório RPMS contem aproximadamente 1 GB de arquivos. Esses arquivos são compactados e assume-se que os pacotes precisem de 2 a 3MB de espaço para cada 1MB no pacote RPM. Se o pacote RPM é de 6MB, por exemplo, você vai precisar entre 12 e 18MB de espaço para instalá-lo.

Personalizar quais pacotes estão disponíveis para instalação é uma opção quando instalando o sistema via FTP, NFS e disco rígido. Você também pode copiar os arquivos de um CD-ROM num disco rígido e instalar a partir de lá com uma listagem personalizada. Instalações via FTP e NFS só podem ser alteradas se você tem acesso root ao servidor da rede ou se o administrador do sistema quer trabalhar com você. As seguintes situações podem tornar uma instalação personalizada desejável: quando você obtém o Red Hat via FTP com uma conexão lenta ou quando projetando um conjunto de software para ser usado por todas as estações Red Hat numa rede.

Para personalizar uma instalação, você tem que obter o arquivo /base/comps, que contém a lista de pacotes que uma instalação completa normalmente inclui. Depois, faça o download dos pacotes que gostaria de instalar e modifique o /base/comps de sua instalação para refletir a lista de pacotes que você escolheu.

Se você tem pacotes RPM locais, pode adicioná-los no arquivo comps também.

14.2.6.1 O arquivo comps

O programa de instalação do Red Hat usa o arquivo /RedHat/base/comps (o arquivo usado aqui é um exemplo do Red Hat Linux versão 4.0) para determinar quais pacotes estão disponíveis no diretório /RedHat/RPMS em cada categoria a ser instalada. O arquivo é organizado por categoria e cada categoria contém uma lista de pacotes que o Red Hat acredita formarem o mínimo necessário para aquela seção. Nota: somente a parte do nome do pacote (pacote-versão-revisão.rpm) é listada no arquivo. Isso significa que o arquivo comps geralmente pode ser usado de uma versão do Red Hat para a outra. Uma seção nesse arquivo tem a seguinte estrutura:

número categoria pacote

end

Um tag para identificar o número da categoria, a categoria, uma lista de nomes de pacotes e a palavra end para marcar o fim de uma categoria

That is a tag to identify the category number, the category, a list of the package names in the category, and the tag ``end" to mark the end of the category.

Sem exceções, todos precisam dos pacotes de software listados na seção Base do arquivo. As outras seções geralmente podem ser personalizadas ou eliminadas para satisfazer uma necessidade particular. Por exemplo, há três tipos de Networked Stations: "plain", management, e dial-up. Um exame destas seções mostra que muitos pacotes de software estão listados nas três categorias, mas alguns pacotes de software são específicos para cada categoria. Se você estiver criando uma estação Dial-up Networked, seguramente pode eliminar as seções "plain" e "management" e quaisquer pacotes únicos para essas categorias. Reciprocamente, se você só precisa a capacidade básica para estações de trabalho (workstations) de rede, as outras seções podem ser eliminadas do arquivo assim como o software único a essas seções. Você precisa certificar-se que tenha todos os pacotes listados nessa categoria. Se você tiver pacotes personalizados exclusivos (não fornecidos pela Red Hat), você deve adicioná-los a uma categoria existente que seja adequada e não criar uma nova categoria.

Como a lista de pacotes em cada categoria só contém o nome do pacote (e não o nome inteiro, como pacote-versão-revisão.rpm), você pode substituir qualquer atualização que tenha sido disponibilizada pela Red Hat no diretório updates. O programa de instalação é relativamente insensível à versão. A única precaução é garantir que as dependências sejam satisfeitas. Quando um pacote RPM é construído, o RPM mesmo tenta determinar quais pacotes devem ser instalados para que esse novo funcione (o desenvolvedor RPM também tem controle direto disto - ele pode adicionar dependências que o RPM não detecte). Aqui é onde um pouco de experimentação ou pesquisa pode ser necessária. Uma maneira de determinar dependências de pacote, por exemplo, é perguntar ao pacote com o comando rpm:

rpm -q -p -R bash-1.14.7-1.i386.rpm libc.so.5 libtermcap.so.2

A opção "-q" põe o rpm em modo pergunta, a "-p" faz com que o rpm verifique um pacote não instalado, e "-R" diz ao rpm para listar as dependências do pacote. Neste exemplo, libc.so.5 e libtermcap.so.2 são necessárias. Como libc e termcap são necessários para o bash, você tem que assegurar-se que os pacotes libc e libtermcap estejam presentes para instalar o bash. Desde que você instale todo o sistema base, você pode inicializar o sistema quando a instalação terminar. Você pode adicionar pacotes extras ao Red Hat quando necessário mesmo que o programa de instalação relate que a instalação de um pacote falhou porque suas dependências não foram respeitadas.

A tabela 2.4 descreve as categorias de software encontradas no arquivo comps (Red Hat v4.0)

RPM Category	Required?	Comments
BASE	Yes	Should not be customized
C Development	Highly Recommended	Need the minimal system to compile a kernel
Development Libs	Highly Recommended	Need the minimal system to compile a kernel
C++ Development	Optional	C++ Development
Networked Workstation	Recommended; Required for other network software	Whether you are on an Ethernet or going to dialup networking, you need to install this package suite; You shouldn't customize this.
Anonymous FTP/Gopher Server	Optional	If your Linux box is going to serve files via FTP or Gopher.
Web Server	Optional	Useful for Web Developers for local development, required if you serve Web Pages.
Network Management Workstation	Optional	Has additional tools useful for dialup as well as Ethernet network
Dialup Workstation		Required if you are going to dialup.
Game Machine	Recommended	Need I say more? Fortunes are required for
	Optional	humor. If you have supported hardware.
Multimedia Machine	Optional	If you want to run X
X Window System	Optional	If you have supported hardware.
X Multimedia Support	Optional	Installation of the entire package is
TEX Document Formatting	Optional	recommended. The One True Editing Environment.
Emacs		Requires X.
Emacs with X Support	Recommended	Huh?
MS-DOS and Microsoft	Recommended	
Windows Connectivity	Optional	Manual pages should always be installed.
Extra Documentation	Required	

Table 2.4: Pacotes importantes do Red Hat Linux.

14.2.7 Instalação mínima recomendada

É difícil determinar quanto espaço em disco uma instalação requer. Entretanto, alguém instalando via FTP deveria pegar o sistema Base e o suporte para dialup e instalar somente isso. Pacotes adicionais podem ser obtidos e adicionados quando houver necessidade. Claro, se você quer programar em C, deve pegar os pacotes relevantes e editar o arquivo comps apropriadamente.

Se, durante a instalação, você encontrar um pacote que requeira outro pacote que você não tem, ou fizer um erro no arquivo comps, geralmente pode terminar a instalação e ainda ter um sistema inicializável. Você pode corrigir o problema adicionado manualmente o pacote que falhou e suas dependências

posteriormente. Em geral, pegue o sistema **Base** inteiro e uma das opões de rede (**Dialup Support**, por exemplo), e você poderá adicionar qualquer outra coisa mais tarde.

14.2.8 Quanto espaço você realmente precisa?

A tabela 2.5 apresenta o espaço em disco necessário para o Red Hat Linux (versão 4.0) e vários subsistemas.

Uso da Partição	Recomendado	Comentários
Swap	2 x memória RAM	
Root system, sem o X	100 a 200 MB	Dependendo de ferramentas, como compiladores, que forem necessários
Root sytem, com o X	250 a 350 MB	Dependendo das ferramentas instaladas
/home	5 a infinitos MB	Depende do número de usuários.
/var	5 a infinitos MB	Depende do número de usuários.
/usr/local	25 a 200 MB	Usado por programas não no formato RPM.
/usr	350+ MB	

Table 2.5: Requisitos de espaço em disco típicos do Red Hat Linux v4.0.

14.2.9 Instalação

Até agora, você já deve ter seu kit de disquetes de instalação criado, preparado seu disco rígido e ter sua mídia de instalação pronta. Os detalhes da instalação estão adiante. A primeira coisa é inicializar seu sistema e configurar o programa de instalação para instalar a partir da mídia selecionada. Depois disso, a instalação segue com os mesmos passos para todos. Você precisa reiniciar o computador com o disquete chamado "Disquete de Inicialização" (Boot Disk).

14.2.10 Ainda sobre a mídia de instalação

Quando o disquete de inicialização inicia, o kernel tenta detectar qualquer hardware para o qual tenha drivers compilados. Uma vez que a inicialização é completada, uma mensagem aparece perguntando se você tem um monitor colorido (se tiver, selecione "ok"). Depois vem a tela de Boas Vindas do Red Hat. Escolha "ok" para continuar. A próxima pergunta é se você precisa de suporte PCMCIA. Você deve responder "yes" (sim) se estiver instalando num laptop - insira o disque suplementar quando solicitado. Depois disso, é apresentada uma tela que pede qual tipo de método de instalação deve ser usado. Siga as instruções nas próximas seções.

14.2.10.1 Instalação por CD-ROM

Para instalar de um CD-ROM, marque a opção "Local CD-ROM" da lista e clique "ok". O programa perguntará se você tem um SCSI, IDE/ATAPI ou CD-ROM proprietário.

Se você escolher SCSI, terá de saber que tipo de placa SCSI você tem. O programa mostrará uma lista. Encontre sua placa na lista. Depois de selecioná-la, o programa perguntará se você quer um "AUTOPROBE" (detecção automática) ou "SPECIFY OPTIONS" (especificar opções). A maioria escolheria AUTOPROBE, o que faz o programa procurar por sua placa SCSI e habilitar suporte para ela quando encontrada.

Depois do programa ter localizado o CD-ROM do Red Hat com sucesso, leia a próxima seção.

NOTA: Muitos computadores atualmente podem inicializar automaticamente do CD-ROM. Se você tem uma máquina assim você pode inicializar diretamente com o CD-ROM do Red Hat sem usar os disquetes de inicialização. Depois de inicializar, você poderá instalar o sistema a partir do CD-ROM, mas inicializando do CD é equivalente a inicializar com o disquete; qualquer driver adicional ainda pode ser necessário.

14.2.10.2 Instalação a partir do disco rígido

Para instalar do disco rígido, selecione essa opção e escolha "ok".

14.2.10.3 Instalação por NFS

Para instalar por NFS, selecione esta opção e escolha "ok". Você deve escolher a placa Ethernet instalada na máquina para que o programa possa carregar o driver correto. Marque a placa adequada na lista e selecione "ok", permitindo um "AUTOPROBE" na sua placa.

Se sua máquina trancar, você deve pressionar Ctrl-Alt-Delete para reiniciar o sistema. Grande parte das vezes, quando isso ocorre é porque a detecção tentou se comunicar com uma placa não ethernet. Se acontecer, tente novamente com a opção "SPECIFY OPTIONS" e forneça os dados sobre sua placa na forma:

ether=IRQ,IO PORT,eth0

Isso instrui o programa a olhar na localização especificada pelos valores de IRQ e IO_PORT da placa Ethernet. Se sua placa Ethernet é configurada para IRQ11 e I/O 0x300, especifique:

ether=11,0x300,eth0

Nota: isso não é necessário para placas plug and play, como a maioria das placas PCI.

Depois que a placa foi encontrada com sucesso, você terá que fornecer informação de TCP/IP sobre sua máquina e sobre o servidor NFS com os pacotes de instalação. Primeiro você terá que fornecer o endereço IP da máquina a ser instalada, a máscara de rede, o gateway padrão, e o servidor de DNS primário. Por exemplo:

IP Address: 192.168.181.21

Netmask: 255.255.255.0

Default Gateway: 192.168.181.1

Primary Nameserver: 192.168.181.1

Depois de selecionar ok, terá de fornecer o nome de domínio e o nome da máquina. Por exemplo, se seu nome de domínio é infomagic.com e seu computador se chama vador, entre:

Domainname: infomagic.com

Host name: vador.infomagic.com

Secondary nameserver IP: servidor DNS secundário - entre se

necessário

Tertiary nameserver IP: servidor DNS terciário - entre se

necessário

A última tela pede o servidor NFS e o nome do diretório exportado contendo a distribuição Red Hat. Se o seu servidor NFS é, por exemplo, redhat.infomagic.com, entre:

NFS Server name: redhat.infomagic.com
Red Hat Directory: /pub/mirrors/linux/RedHat

Se você desconhece esses dados, pergunte ao seu administrador de sistema. Depois de entrar esses valores, selecione ok para continuar. Se o programa de instalação relata um erro ao localizar a distribuição do Red Hat, esteja certo de ter entrado os valores corretos nos campos acima e que o administrador de sua rede tenha dado permissões de exportação corretas para a sua máquina.

14.2.10.4 Instalação por FTP

A instalação por FTP é similar à instalação por NFS descrita acima. Você é questionado sobre sua placa Ehternet e a informação TCP/IP de sua máquina. Entretanto, o programa também perguntará sobre o site FTP e o diretório onde está o Red Hat ao invés do servidor NFS. Uma nota sobre fazer uma instalação FTP: procure o site FTP mais próximo e menos ocupado.

14.2.11 Passando pelo resto da instalação

- No próximo passo você terá de informar se está instalando um Novo Sistema ou Atualizando de outro antigo. Se estiver atualizando, não poderá reparticionar o disco ou configurar qualquer coisa no sistema que não seja o LILO. Selecione "INSTALL" ou "UPGRADE" para continuar.
- 2. Se você estiver atualizando, deverá informar a partição root do seu sistema Red Hat atual. Selecione a partição apropriada e pressione ok. Se você está instalando pela primeira vez, você precisa particionar seu disco com o espaço determinado acima.
- Depois de criar as partições de Linux Nativa e de Swap, você deve reiniciar e habilitar a partição de swap. O programa irá perguntar em qual partição você quer instalar o Linux. Se for uma atualização,

selecione a partição root. Você tem que configurar e escolher pelo menos uma partição, que será sua partição root. Selecione a partição root e então (a menos que esteja atualizando) aparecerá uma tabela de partições disponíveis. Escolha as partições apropriadas e a opção "EDIT" para indicar quais partições serão usadas para cada diretório. Se tiver mais de uma partição para a instalação do Linux, agora é a hora de designá-las.

- 4. Depois, uma lista de categorias de software a serem instalados é apresentada, seguida de uma chance de personalizar quais pacotes de cada categoria devem ser instalados. Se você não tiver instalado o Red Hat ou outra distribuição Linux antes, simplesmente escolha as categorias que quiser e deixe o programa de instalação instalar o padrão para cada categoria. Se precisar de um pacote que não foi instalado originalmente, você sempre pode instalá-lo no futuro. Enquanto o software estiver instalando, você vai ver um indicar de progresso e deveria tomar um ou dois cafés. A instalação pode levar 30 min a uma hora ou até mais, dependendo de sua configuração de hardware e escolha de pacotes.
- 5. Depois da instalação feita, você poderá configurar seu mouse.
- 6. A seguir vem a configuração do sistema X Window. É recomendado que você espere até que inicialize o sistema pela primeira vez para iniciar a configuração do X. Se algo sair errado com a configuração do X, você pode ter que reiniciar todo o processo de instalação se o programa de instalação não conseguir se recuperar.
- 7. Se você não tem uma placa Ethernet, não configure sua rede agora. Se você tem uma placa de rede e não a configurou antes, você deve configurá-la agora. Configuração para acesso discado deve ser feita após completar a instalação.
- 8. A seguir você deve configurar o relógio do sistema. UTC (Coordinated Universal Time) é uma boa escolha se você estiver numa rede e quiser que horários de verão sejam configurados adequadamente. "Local Time" está ok se você tem um computador isolado.
 - 9. Se você não tem um teclado US, terá de especificar a configuração do seu teclado.
 - 10. O sistema vai pedir pela senha de root. Não a esqueça. Recuperar uma senha de root não é um problema trivial. Você vai precisar da senha para acessar o sistema quando inicializá-lo pela primeira vez.
 - 11. Finalmente virá a configuração do LILO.

Se você não tiver instalado uma partição root que inicie e termine entre os cilindros 0-1023, *não instale o LILO*. Se houver um problema na inicialização do sistema, use o disquete de inicialização do Windows e no prompt A: entre FDISK /MBR. Isso vai permitir que você inicialize o sistema no Windows como fazia antes do LILO ser instalado. Você então poderá usar o disquete de inicialização do Red Hat com os seguintes parâmetros no prompt de inicialização para iniciar o sistema pelo disco rígido:

boot: rescue root=/dev/particao-root ro load_ramdisk=0

Onde *particao-root* é a partição root do sistema. Depois que a instalação estiver completa, você estará pronto para reiniciar o sistema e usar o Linux.

14.2.12 Após a instalação

Agora que você já instalou e reiniciou o sistema com o Linux pela primeira vez, há algumas dicas úteis a conhecer.

14.2.12.1 Entendendo o prompt do LILO

Quando você liga ou reinicia o sistema, você pode ver o prompt do LILO, que deve estar configurado para um intervalo de 30 segundos antes que ele inicie automaticamente. Quando o LILO aparece na tela, se você não fizer nada, o sistema operacional padrão irá inicializar no tempo ajustado. Entretanto, a partir do LILO você pode controlar vários aspectos de como o Linux inicializa ou ainda indicar um outro sistema operacional para que o LILO inicialize. Se você quiser cancelar o comportamento padrão do LILO, apertando Shift quando ele aparecer vai causar o aparecimento do prompt "boot:". Pressionado Tab, uma lista de sistemas operacionais disponívies vai aparecer:

LILO boot:

dos linux

boot:

Isso indica-nos que o "dos" é o sistema operacional padrão, que vai inicializar se nada for digitado. Para inicializar o Linux, digite "linux". Entretanto, o LILO permite que você passe parâmetros ao kernel do Linux que alteram seu comportamento padrão. Você pode estar experimentando com arquivos de configuração e fez algo que impediu o sistema de iniciar, por exemplo. Se isso aconteceu, você quer iniciar o sistema até o ponto que ele lê os arquivos de configuração e nada além disso. O parâmetro para isso é "single":

boot: linux single

Inicializa o sistema no modo mono-usuário para que você possa tomar a ação corretiva. Isso também é útil se o seu sistema não inicializa por qualquer motivo.

14.2.12.2 Logando no sistema pela primeira vez

Agora que você está na frente do prompt de login: pela primeira vez, pode estar imaginando como entrar no sistema. Neste momento, num sistema recém instalado, há somente uma conta para entrar - a conta administrativa "root". Essa conta é usada para gerenciar o sistema e fazer outras coisas, como configurá-lo, adicionar ou remover usuários, software, etc. Para entrar na conta, digite "root" no prompt "login:" e pressione <Enter>. O sistema pedirá a senha que você escolheu durante a instalação. Entre a senha no prompt "password:". O prompt do sistema "[root@localhost] #" aparece depois você ter tido sucesso no login. O prompt do sistema diz a você duas coisas: você está logado como "root" e, neste caso, sua máquina se chama "localhost". Se você

deu nome a sua máquina durante o processo de instalação, esse nome aparecerá no lugar de "localhost".

14.3 Slackware

Esta seção sobre o Slackware foi escrita por Sean Dreilinger.

14.3.1 O Slackware não é para você. (Ou tal vez seja.)

Bem vindo à distribuição Slackware do Linux! Esta seção tem por objetivo ajudar o novo usuário ou administrador Linux a avaliar o Slackware, planejar um sistema Slackware e instalar o Slackware Linux.

Escolher ou não o Slackware é uma decisão difícil. Pode parecer trivial agora, mas os ambientes Linux têm a tendência de irem acumulando mais e mais responsabilidade nos ambientes computacionais. Muitos Linux experimentais evoluíram no seu primeiro ano para se tornarem máquinas de missão-crítica servindo muito mais usuários e propósitos do que inicialmente planejado. Slackware é uma das distribuições mais usadas. Quando se trata de encontrar a distribuição mais nova, fácil ou mais cuidadosamente planejada do Linux, podemos dizer que o Slackware se encaixa em "nenhuma das respostas acima". Uma retrospectiva sobre a vida e evolução do Slackware nos esclarece a questão.

14.3.2 Uma breve história

Em 1993, a Soft Landing System (SLS) criou uma das primeiras distribuições organizadas do Linux. Embora fosse um grande passo, a distribuição da SLS tinha muitos defeitos (na verdade nem funcionava, para iniciantes). O Slackware, uma dádiva de Deus enviada por Patrick Volkerding, resolveu a maioria dos problemas, foi copiado via FTP e impresso em CD-ROMs em todo o mundo, e rapidamente se tornou a distribuição mais usada do Linux. Por algum tempo, o Slackware foi a única "solução" completa do Linux. Outros mantenedores, tanto comerciais como sem fins lucrativos, gradualmente desenvolveram distribuições que valem a sua consideração.

Em Janeiro de 1994, o Slackware atingiu tamanha expansão que ganhou notoriedade popular normalmente reservada para pop stars. Uma fofoca se espalhou pela Usenet sugerindo que o projeto Slackware era trabalho de bruxas e adoradores do diabo!

"Linux, o sistema livre... exceto pela sua ALMA! MOUHAHAHAHA!"

14.3.3 Por quê, então?

Se você é um administrador de sistema, pode ser que já esteja lidando com um ou mais servidores chave usando o Slackware. A menos que você tenha tempo de experimentar no trabalho, aderir a uma distribuição já amplamente testada é o caminho mais fácil a seguir. Se você espera conseguir ajuda de amigos ou colegas letrados em UNIX, melhor ter certeza

que eles estejam rodando algo compatível - e é provável que estejam usando Slackware. Seus defeitos são amplamente conhecidos, em sua maioria já descobertos, documentados e corrigidos quando possível. Você pode juntar um Slackware, tapar os furos conhecidos, e instalar algumas ferramentas complementares de outras distribuições Linux para criar um excelente servidor UNIX ou estação de trabalho, tudo em meio dia de trabalho.

14.3.4 Atualização? Pense duas vezes!

24-Aug-95 NOTE: Tentar atualizar do Slackware ELF para o Slackware a.out irá, com certeza, causar todo o tipo de problemas. Não tente isso.

Patrick Volkerding

Uma coisa que você não escuta freqüentemente no Slackware é atualização. Os programas de configuração do Slackware foram planejados para instalar um sistema novo em discos rígidos vazios ou partições vazias. Instalar um Slackware sobre uma instalação antiga pode apagar suas aplicações personalizadas e causar problemas de compatibilidade entre aplicações atualizadas e arquivos antigos no mesmo sistema. Quando o Slackware foi montado, todo o mundo era um novo usuário Linux, e o sistema era sempre experimental - reinstalar o sistema operacional inteiro e as aplicações era a norma. Hoje, muitas instituições e empresas rodam aplicações de missão-crítica num Slackware. Nesse ambiente, uma simples reinicialização é uma atividade planejada e colocar o sistema abaixo e sobreescrever os arquivos de usuários e aplicações personalizadas é absolutamente inaceitável.

Ensinar como enganar um Slackware para atualizá-lo está além do escopo deste capítulo, mas é praticável se você for um administrador UNIX experimentado e tiver tomado as precauções necessárias para preservar as modificações e arquivos de usuários.

Você pode ler, chorar e aprender a partir da experiência de Greg Louis no seu mini HOWTO "Upgrading Your Linux Distribution", disponível onde houver um mirror do LDP (Linux Documentation Project). A home page do LDP é:

http://www.linuxdoc.org

14.3.5 Selecione um método de instalação

O Slackware pode ser instalado a partir de meios variados, inclusive a rede. Todos os métodos requerem que você tenha no mínimo dois disquetes para dar início ao processo.

14.3.6 Discos de inicialização: sempre uma boa idéia

Mesmo que você tenha uma conexão T-3 com a Internet, que permite que você sugue uma distribução do Slackware imediatamente da rede, é melhor começar criando os dois discos de inicialização do Slackware (boot e root) antes de iniciar. Se ocorrer um acidente (falta de luz, amigos maliciosos metendo a mão no teclado, ou mesmo um erro humano), esses dois disquetes

podem reviver seu sistema ou ao menos recuperar seus arquivos pessoais.

14.3.7 Planilha para in stalação do Slackware

Depois de copiar todos os arquivos, o Slackware pode continuar e fazer a maioria das configurações de rede e sistema, se você estiver pronto. Para ajudá-lo a planejar suas decisões, esta seção consiste de uma planilha derivada do programa de instalação do Slackware. Se puder usar esta planilha para guardar suas respostas antecipadamente (enquanto seu computador ainda está trabalhando!), estará pronto com os detalhes necessários - partições, endereços IP, IRQ do modem e mouse, nomes do computador e de domínio, e outros que você terá de fornecer durante a instalação.

1. Teclado: O programa de instalação do Slackware setup vai querer saber se você precisa remapear o teclado para algo diferente do padrão USA 101 teclas?

sim ou não

2. Configuração do Swap: Você já tem uma partição preparada com tipo 82 (Linux Swap)?

sim ou não

Você quer que o programa de instalação use o mkswap nas suas partições de swap? Normalmente "sim", a não ser que já tenha feito isso antes.

sim ou não

3. Preparando a Partição Principal do Linux: O setup vai listar todas as partições do tipo 83 (Linux Native) e perguntar qual você quer usar como partição root (/). Use o formato do tipo /dev/hda3 ou como o dispositivo for chamado.

nome da partição

Última chance de recuar! Quando usar a opção "instalar do início", você tem que instalar para uma partição em branco. Se ainda não a tiver formatado, você terá de fazê-lo quando solicitado. Entre "i" para Instalar do início ou "a" para adicionar novos programas ao seu sistema existente.

[i]nstalar or [a]dicionar

(Re)formatar a partição principal do Linux. Você quer formatar esta partição?

sim, não ou checar setores também

O sistema de arquivos ext2fs tem por padrão um inode (apontador para arquivo) para cada 4096 bytes de espaço. Se você planeja ter muitos pequenos arquivos no seu disco, você pode precisar mais inodes (vai precisar de um para cada arquivo). Você pode mudar a densidade para um inode por 2048 bytes ou mesmo um para cada 1024 bytes.

4096 (default). 2048, or 1024

4. Preparando Partições Linux Adicionais: Você pode montar outras

partições para o /usr ou /usr/X11 ou o que quer que seja . Você quer usar alguma outra partição Linux para montar alguns de seus diretórios?

sim ou não

Estas são suas partições Linux (lista será mostrada). Estar partições já estão em uso (outra lista mostrada). Entre a partição que gostaria de usar, ou digite "q" (quit) para sair. Use o formato como /dev/hda3 ou qualquer que seja o nome de seu dispositivo.

Nome da partição ou [q]uit

Quer formatar esta partição?

sim, não ou checar setores também

Agora esta nova partição precisa ser montada em algum ponto de sua árvore de diretórios. Se quiser colocá-la sob /usr/X11R6, por exemplo, responda /usr/X11R6. Onde quer montar esta nova partição?

Ponto de montagem

Quer montar algumas partições adicionais?

sim ou não

5. Configuração de partições DOS e OS/2: As seguintes partições DOS FAT ou OS/2 HPFS foram encontradas (lista de partições mostrada). Você quer configurar alguma dessas partições para ser visível a partir do Linux?

sim ou não

Por favor entre a partição que gostaria de acessar do Linux, ou digite "q" (quit) para sair. Use o formato como /dev/hda3 ou qualquer que seja o nome de seu dispositivo.

Nome da partição ou [q]uit

Agora esta nova partição deve ser montada em algum lugar na sua árvore de diretórios. Por favor entre o diretório sob o qual gostaria de colocá-la. Você pode escolher /dosc ou /dosd ou qualquer coisa que deseje. Onde quer montar esta partição?

Ponto de montagem

6. Seleção de Mídia:

- (a) Install from a hard drive partition
- (b) Install from floppy disks.
- (c) Install via UFS.
- (d) Install from a pre-mounted directory.
- (e) Install from CD-ROM.
- 7. Instalando de uma partição do disco rígido: Para instalar diretamente de um disco rígido, você tem que ter uma partição contendo a distribuição Slackware com um subdiretório para cada disco. Se a distribuição está em /stuff/slack, por exemplo, você vai precisar de diretórios /stuff/slack/a1, /stuff/slack/a2, e assim por diante, cada um

contendo os arquivos que estariam naquele disco. Você pode instalar de partições DOS, HPFS ou Linux. Entre a partição onde se encontram as fontes do Slackware, ou pressione "p" para listar as partições.

Nome da partição ou lista de [p]artições

Em que diretório nesta partição estão os fontes do Slackware? No exemplo acima, seria /stuff/slack.

Nome do diretório

Que tipo de sistema a partição que contém a distribuição contém?

```
(a) FAT (MS-DOS, DR-DOS, OS/2)
(b) Linux Second Extended File System
(c) Linux Xiafs
(d) Linux MINIX
(e) OS/2 HPFS
```

8. Instalando de um diretório pré-montado: Okay, nós vamos instalar a partir de um diretório que já está montado. Ele pode estar montado normalmente ou por NFS. Você tem que especificar o nome do diretório que contém os subdiretórios com os arquivos da distribuição. A partir de qual diretório você quer instalar?

Nome do diretório

9. Instalando de disquetes: As séries básicas do Slackware (A) podem ser instaladas a partir de disquetes 1.44MB.

The base Slackware series (A) can be installed from 1.2M or 1.44M media. Most of the other disks will not fit on 1.2M media, but can be downloaded to your hard drive and installed from there later. Which drive would you like to install from (1/2/3/4)?

```
/dev/fd0u1440 (1.44M drive a:)
/dev/fd1u1440 (1.44M drive b:)
/dev/fd0h1200 (1.2M drive a:)
/dev/fd1h1200 (1.2M drive b:)
```

10. Instalando via NFS: Esta máquina está atualmente rodando na rede da qual você quer instalar? Se está, não vamos tentar reconfigurar sua placa ethernet. Sua rede está rodando?

sim ou não

Você precisa entrar o endereço IP que você quer atribuir a esta máquina. Exemplo: 111.112.113.114. Qual seu endereço IP?

Endereço IP

Agora precisamos saber sua máscara de rede. Tipicamente seria 255.255.255.0. Qual é sua máscara de rede?

Máscara de rede

Você tem um gateway?

sim ou não

Qual o endereço do seu gateway?

Endereço IP

Bom! Estamos no final, mas precisamos saber onde estão os pacotes a instalar. Primeiro, precisamos do endereço IP da máquina onde estão os fontes do Slackware. Como você já está na rede, deve ser capaz de usar o nome da máquina ao invés do endereço IP. Qual o endereço IP do seu servidor NFS?

Endereço IP

Deve existir um diretório no servidor com os fontes do Slackware com subdiretórios para cada disco. O setup precisa saber o nome do diretório que contem os subdiretórios no servidor. Por exemplo, se seu disco A3 está em /slackware/a3, você deve responder /slackware. Qual o diretório?

Nome do diretório

11. Instalado deCD-ROM: Qual tipo de CD-ROM você tem?

- Works with most ATAPI/IDE CD drives (/dev/hd*) # trabalha (a) com IDE
- (b) SCSI (/dev/scd0 ou /dev/scd1)
- Sony CDU31A/CDU33A (/dev/sonycd) (c)
- (d) Sony 531/535 (/dev/cdu535)

Try to scan for your CD drive # tentar procurar o CD (m)

a, b, c, d, e, f, g, h, i, j, k, l, m

CD-ROM IDE: Entre o nome do dispositivo que representa seu disco de CD-ROM. Provavelmente será um destes (em ordem do mais ao menos provável): /dev/hdb /dev/hdc /dev/hdd /dev/hde /dev/hdf /dev/hdg /dev/hdh /dev/hda

Nome do dispositivo

CD-ROM SCSI: Qual CD-ROM SCSI você está usando? Se não estiver certo, use /dev/scd0.

1. /dev/scd0

2. /dev/scd1

método de instalação: Com o CD do Slackware, você pode rodar a maior parte do sistema a partir dele, se estiver com pouco espaco em disco ou quer apenas testar o Linux sem passar por uma instalação completa. Qual tipo de instalação você quer (slackware ou slacktest)?

slakware

Instalação normal no disco rígido

slaktest

Linca /usr->/cdrom/live/usr para rodar principalmente do CD.

slakware or slaktext

- **12. Seleção das Séries:** Identifique que pacotes você planeja instalar. Você deve especificar uma combinação de discos. Por exemplo, para instalar o sistema base, o X Window e o Tcl Toolkit, você entraria : a x tcl. Que conjuntos quer instalar?
 - A Sistema Linux Base
 - AP Vários aplicativos que não precisam do X
 - D Utilitários para Desenvolvimento (C, C++, fontes do kernel, Lisp, Perl, etc)
 - E GNU Emacs
 - F listas de FAQ
 - K Fontes do Kernel do Linux
 - N Rede (Networking) TCP/IP, UUCP, Mail
 - Q Kernels extras com drivers especiais (necessários para CDs não SCSI)
 - T TeX
 - TCL Tcl/Tk/TclX, linguagem Tcl, e toolkit Tk para desenvolver aplicações X
 - X sistema X Window
 - XAP Aplicações para o X
 - XD sistema de desenvolvimento para o servidor X11
 - XV XView (Gerenciador de Janelas OpenLook e aplicativos)
 - Y Jogos (que não usam o X)

Qualquer combinação de a ap d e f k n q t tcl x xap xd xv y e outros conjuntos oferecidos, separada por espaços.

13. Instalação do Software: A seguir, os pacotes de software serão transferidos para o seu disco rígido. Se esta é a primeira vez que estiver instalando o Linux, provavelmente usará o modo PROMPT. No modo prompt você poderá pedir a instalação de outros pacotes. Se você não usar o modo prompt, o programa de instalação irá adiante e instalará tudo que você selecionou sem perguntar. Você quer usar o modo prompt?

sim ou não

14.Configuração Extra: Se você quiser, pode passar pelas opções e reconfigurar seu harware, criar um disco de inicialização e instalar o LILO. Se tiver instalado uma nova imagem do kernel, você deve passar por estes passos novamente. De outra forma, você decide.

sim ou não

15.Criação do Disquete de Inicialização: É recomendado que você crie um disco de inicialização. Quer fazer isto?

sim ou não

Insira um disco formatado no drive. Esse será seu disco de inicialização. Use-o para inicializar o Linux até que o LILO tenha sido configurado para inicializar de seu disco rígido. Qualquer dado no disquete será destruído. Insira o disco e pressione <Enter>, ou "s" se quiser saltar esta etapa.

<Enter> ou [s]altar

16. Configuração do Modem: Um link em /dev será criado do seu dispositivo de discagem (cua0, cua1, cua2 ou cua3) para /dev/modem. Você pode mudar esse link depois se colocar o modem em outra porta. Quer configurar seu modem?

sim ou não

Estes são as portas seriais padrão para os dispositivos E/S. Em qual seu modem está ligado (0, 1, 2 ou 3)?

- /dev/ttyS0 (ou COM1: no DOS)
 /dev/ttyS1 (ou COM2: no DOS)
 /dev/ttyS2 (ou COM3: no DOS)
 /dev/ttyS3 (ou COM4: no DOS)
- 0. 1. 2. or 3
- 17. Configuração do Mouse: Um link será criado em /dev do seu mouse para /dev/mousr. Você pode mudar esse link posteriormente se mudar o mouse de porta. Quer configurar seu mouse?

sim ou não

Estes tipos são suportados. Que tipo de mouse você têm?

- (a) compatível com serial Microsoft
- (b) Quickport ou PS/2
- (c) Logitech Bus Mouse
- (d) ATI XL Bus Mouse
- (e) Microsoft Bus Mouse
- (f) Mouse Systems serial mouse
- (g) Logitech serial mouse

a, b, c, d, e, f ou g

Estas são as portas padrão de E/S. Em qual seu mouse está conectado?

/dev/ttyS0 (ou COM1: no DOS)/dev/ttyS1 (ou COM2: no DOS)/dev/ttyS2 (ou COM3: no DOS)

3 /dev/ttyS3 (ou COM4: no DOS)

0. 1. 2. or 3

18.Configuração de Rede: Agora vamos tentar configurar seu mail e TCP/IP. Esse process possivelmente não vai funcionar em todas as configurações de rede possíveis, mas deve ser um bom começo. Você poderá reconfigurar seu sistema a qualquer tempo chamando netconfig. Primeiro, precisamos do nome que quer dar para sua máquina. Somente o nome base é necessário agora (não o nome de domínio). Entre o nome do host (máquina):

Nome do Host

Agora precisamos do nome do domínio. Não deve iniciar com ".". Entre o nome do domínio.

Nome do domínio

Se você planeja usar TCP/IP através de loopback, então seu endereço IP será 127.0.0.1, e podemos pular uma série de perguntas. Planeja usar *somente* loopback?

sim ou não

Entre o endereço IP para a máquina local. Exemplo: 111.112.113.114.

Endereço IP

Entre o endereço do gateway (exemplo: 111.112.113.1). Gateway é a máquina responsável por trocar dados com outras máquinas fora de sua rede. Se você não tem um gateway agora, pode editar o /etc/rc.d/rc.inet1 posteriormente , ou até mesmo colocar seu próprio endereço IP aqui. Entre o endereço IP do gateway:

Endereço IP

Entre sua máscara de rede. Geralmente deve ser algo como: 255.255.255.0. Entre a máscara:

Máscara de rede

Você vai acessar um servidor de nomes (DNS)?

sim ou não

Por favor indique o endereço IP do servidor. Você pode adicionar outros Servidores de Nome editando /etc/resolv.conf. Qual o servidor de nomes do seu domínio?

Endereco IP

Agora você pode reinicializar seu computador pressionando Ctrl-Alt-Del. Se você instalou o LILO, remova o disco de inicialização antes de reiniciar. Não esqueça de criar seu /etc/fstab se não tiver um!

14.3.8 Fazendo o Slackware acontecer

Se você separou um tempo para organizar e planejar como recomendado nas seções anteriores, a instalação mesmo será um abraço. Não há muito o que escrever para explicar o processo real de instalar o

Slackware no seu computador. Siga os passos para criar os disquetes de boot e root e responda a longa série de perguntas feitas pelo programa de instalação. Se você completou a planilha, essas perguntas serão familiares e tudo correrá tranquilamente.

14.3.9 Criando alguns discos de boot

14.3.9.1 Escolha seu kernel!

Para instalar o Slackware Linux, você vai precisa criar um disquete de boot com um kernel especialmente preparado para reconhecer seu sistema. Para instalar o Slackware de um CD-ROM num disco SCSI, por exemplo, o kernel que você colocar no disquete de boot precisa ter os drivers para sua placa SCSI e para seu disco de CD-ROM.

Os kernels são armazenados como arquivos binários compactados que você pode acessar de quase todos os sistemas operacionais para criar o disquete de Boot do Slackware. No site de FTP da Slackware, ou no CD-ROM, você vai encontrar um subdiretório chamado bootdsks.144 contendo imagens de kernel de1.44 MB para criar os disquetes de boot de 1.44MB (3,5").

As tabelas 2.6 e 2.7 dão uma rápida referência das imagens de kernel disponíveis no momento que este livro foi escrito. Informações atualizadas sobre os disquetes de boot podem ser encontradas em:

ftp://ftp.slackware.org/pub/slackware/slackware/bootdsks.144/README. TXT

File	IDE Slackware bootdisks	
aztech.i	CD-ROM drives: Aztech CDA268-01A, Orchid CD-3110, Okano/Wearnes CDD110, Conrad TXC, CyCDROM CR520, CR540	
bare.i	IDE support only.	
cdu31a.i	Sony CDU31/33a CD-ROM.	
cdu535.i	Sony CDU531/535 CD-ROM.	
cm206.i	Philips/LMS cm206 CD-ROM with cm260 adpter card.	
goldstar.i	Goldstar R420 CD-ROM (sometimes sold in a Reveal "Multimedia Kit").	
mcd.i	NON-IDE Mitsumi CD-ROM support.	
mcdx.i	Inproved NON-IDE Mitsumi CD-ROM support.	
net.i	Ethernet support.	
optics.i	Optics Storage 8000 AT CD-ROM (the "DOLPHIN" drive).	
sanyo.i	Sanyo CDR-H94A CD-ROM support.	
sbpcd.i	Matsushita, Kotobuki, Panasonic, CreativeLabs (Sound Blaster).Longshine and teac NON-IDE CD-ROM support.	
xt.i	MFM hard drive support.	

Tabela 2.6: Imagens de disquetes de boot Slackware IDE

File	SCSI/IDE Slackware bootdisks:		
7000fast.s	Western Digital 7000FASST SCSI support.		
Advansys.s	AdvanSys SCSI support.		
Aha152x.s	Adaptec 152x SCSI support.		
Aha1542.x	Adaptec 1542 SCSI support.		
Aha1740.s	Adaptec 1740 SCSI support.		
Aha2x4x.s	Adaptec AIC7xxx SCSI support (For these cards: AHA-274x, AHA-2842, AHA-2940, AHA-2940W, AHA-2944UW, AHA-2944D, AHA-2944WD, AHA-3940, AHA-3940W, AHA-3985, AHA-3985W)		
Am53c974.s	AMD AM53/79C974 SCSI support.		
Aztech.s	All supported SCSI controllers, plus CD-ROM support for Aztech CDA268-01A, Orchid CD-3110, Okano/Wearnes CDD110, Conrad TXC, CyCDROM CR520, CR540.		
Buslogic.s	Buslogic MultiMaster SCSI support.		
Cdu31a.s	All supported SCSI controllers, plus CD-ROM support for Sony CDU31/33a.		
Cdu535.s	All supported SCSI controllers, plus CD-ROM support for Sony CDU531/535.		
Cm206.s	All supported SCSI controllers, plus Philips/LMS cm206 CD-ROM with cm260 adpter card.		
Dtc3280.s	DTC (Data Technology Corp) 3180/3280 SCSI support.		
Eata.dma.s	DPT EATA-DMA SCSI support (Boards like PM2011, PM2021, PM2041, PM3021, PM2012B, PM2022, PM2122, PM2322, PM2042, PM3122, PM3222, PM3332, PM2024, PM2124, PM2044, PM2144, PM3224, PM3334)		
Eata.isa.s	DPT EATA-ISA/EISA SCSI support (Boards like PM2011B/9X, PM2021A/9X, PM2021A, PM2012B, PM2022A/9X, PM2122A/9X, PM2223A/9X)		
Eata.pio.s	DPT EATA-PIO SCSI support (PM2001 and PM2012A).		
Fdomain.s	Future Domain TMC-16x0 SCSI support.		
Goldstar.s	All supported SCSI controllers, plus Goldstar R420 CD-ROM		
In2000.s	Always IN2000 SCSI support		
Iomega.s	IOMEGA PPA3 parallel port SCSI support		

Tabela 2.7: Imagens de disquetes de boot Slackware SCSI.

14.3.10 Entrando em ação

Aqui é o grande anticlímax. Depois de todo o planejamento, preparação e particionamento, você está na reta final. Esteja certo que seu disco de inicialização está no drive e reinicie seu computador. É um bom momento para servir um café e voltar para a máquina pronto para passar uma hora mais ou menos respondendo a perguntas tipo sim ou não.

Entre como root (não tem senha) e digite setup ou setup.tty

14.3.11 O programa de instalação (setup) Slac kware

O Slackware vem com duas versões do programa de instalação. Uma é colorida, baseada em janelas de diálogo, e orientada a menus. A versão alternativa, setup.tty, é baseada em texto e você até pode preferir esta, porque as mensagens de erro ou diagnóstico ficam na tela e não são escondidas pela próxima janela de diálogo. Se você estiver instalando o Slackware num hardware não confiável, eu recomendo fortemente a versão menos colorida (setup.tty). Se você não sabe muito sobre UNIX e se sente mais confortável com uma interface limpa e atrativa para o mesmo processo então, sem sombra de dúvidas, use a versão mais bela.

```
======= Slackware96 Linux Setup (version HD -3.1.0) ========
  Welcome to Slackware Linux Setup.
  Hint: If you have trouble using the arrow keys on your keyboard,
  you can use '+', '-', and TAB instead. Which option would you
like?
______
      HELP
           Read the Slackware Setup HELP file
      KEYMAP
             Remap your keyboard
      MAKE TAGS Tagfile customization program
      TARGET
              Select target directory [now: / ]
      SOURCE
              Select source media
      DISK SETS Decide which disk sets you wish to install
      TNSTAT.T.
              Install selected disk sets
      CONFIGURE Reconfigure your Linux system
      PKGTOOL
              Install or remove packages with Pkgtool
              Exit Slackware Linux Setup
  ___________
  ______
                   < OK >
                             <Cancel>
______
```

Transferir o Slackware a partir daqui para o seu sistema deve envolver pouco mais do que selecionar o que você deseja dos menus. Completando a Seção 3 da planilha adiantadamente, você deve estar apto a passar por cada menu rapidamente até atingir a opção INSTALL (Instalar), ponto em que as coisas podem ficar mais lentas: você foi aconselhado a selecionar a opção PROMPT e ler sobre cada pacote de software, decidindo se ele deve entrar ou

não na sua instalação. A última parte de uma instalação normal é a seção CONFIGURE (Configurar) no menu setup, e as perguntas que você deve responder se assemelham às da segunda metade da planilha da seção 3.

14.3.12 Isso é tudo?

De jeito algum! Neste ponto, ou você está enfrentando algum obstáculo perturbador que não permite a conclusão da instalação, ou, mais provável, está vendo o prompt do root.

darkstar ~#

e perguntando-se: "E agora?"

Bem, se você estiver atormentado por problemas, vai querer ir diretamente para a próxima seção sobre solução de problemas. Se parece que as coisas estão funcionando corretamente, você ainda tem alguns detalhes para ver. É como comprar um carro novo - depois de escolher e pagar pelo carro, ainda há algumas coisas a fazer antes que possa dirigi-lo com segurança - seguro, emplacamento, etc.

14.3.13 Solucionando Problemas de instalação

Nem toda a instalação do Slackware nasce de acordo com a expectativa do administrador do sistema. Eu já enfrentei a situação de ter de atualizar um Slackware depois do trabalho e ficar a noite toda trabalhando para colocar a maldita coisa online até a madrugada, antes que as pessoas chegassem e começassem a reclamar sobre e-mails e news faltando. Esta seção vai passar por alguns problemas comuns da instalação do Slackware, indicando soluções e onde procurar por mais ajuda.

14.3.13.1 FAQs de instalação do Slackware

Patrick Volkerding, o pai do Slackware, tem lidado com muitas questões de novos usuários escutando, respondendo e antecipando perguntas repetidas. Para evitar ter que responder sempre as mesmas pergunta, Patrick criou a documentação e a inlcluiu na distribuição Slackware. Três arquivos que você pode achar muito úteis para suas perguntas iniciais são FAQ.TXT, INSTALL.TXT e BOOTING.TXT.

14.3.13.2 Suporte na Web para o Slackware

A maneira mais fácil de encontrar documentção sobre Linux geralmente é a página do LDP (Linux Documentation Project).

Você também pode encontrar boa documentação na home page do Slackware (http://www.slackware.org).

14.3.13.3 Grupos de News Usenet Para Slackware

A hierarquia comp.os.linux.* é um tesouro de informações sobre Linux,

não necessariamente sobre o Slackware.

14.3.13.4 Listas de discussão sobre Slackware

Veja em http://www.slackware.org/lists/ várias opções de listas de discussão sobre Slackware.

14.3.13.5 Você recebe pelo que paga (suporte comercial)

Suporte comercial de Linux está disponível através de vários consultores e integradores de distribuições.

14.3.14 Não baixe a guarda

Não é hora de contar vantagem, especialmente se seu computador com Slackware será compartilhado ou instalado num ambiente de rede. Cuidar de um computador para uma comunidade ou uso em rede demanda mais do que simplesmente rodar o programa de instalação e esquecer do resto. Nós daremos algumas dicas para tornar seu sistema seguro e compartilhá-lo.

14.3.15 Considere a reinstalação!

Eu sei que você recém passou por uma sessão de instalação longa, e talvez complicada. Mas antes de se mudar para a nova casa, que acabou de construir, considere a possibilidade de destruí-la e começar tudo novamente. Uma citação de Friedrich Nietzsche:

"Um homem aprende o que precisa saber sobre contruir sua casa somente após tê-la terminado."

Se, no processo de instalar o sistema, você teve novas idéias sobre como poderia ter sido diferente, agora é a hora. Se seu Slackware vai ser um computador multi-usuário ou servidor de rede, pode não haver outra oportunidade tão conveniente de reinstalar ou reconfigurar o sistema de maneira radical.

14.3.16 Torne o sistema seguro

14.3.16.1 Saia da rede de uma vez

O Slackware recém instalado é um sistema inseguro. Embora Patrick Volkerding faça o melhor para criar uma distribuição segura, poucos furos inevitáveis tornam-se conhecidos, e *patches* (remendos) ou *workarouds* são disponibilizados para os administradores (e crackers). Se você instalou o Slackware a partir da rede (ex: NFS), você deve temporariamente desconectálo da mesma até que tape alguns furos.

14.3.16.2 Cadastre uma senha de root

Por definição, um novo Slackware não pede senha para o usuário root.

Quando estiver confiante na estabilidade de seu sistema Slackware (depois de algumas horas, não dias ou semanas), crie uma senha para proteger a conta root. Entre como root e digite:

passwd root

14.3.16.3 Crie uma conta para você

Em sistemas grandes, compartilhados, a conta de super-usuário root não é usada como uma conta de trabalho. Se estiver interessado em administrar o sistema ou estiver em uma rede, é um bom conselho a ser seguido. Use o programa /sbin/adduser e crie uma conta própria para entrar no sistema, ao invés de usar a conta root. Eu sempre acho graça quando vejo estudantes e iniciantes orgulhosamente publicando no News como root@minhamáquina.meudomínio. Seja humilde e seguro: crie outra conta de login para seu trabalho diário e use o programa su para entrar como root.

14.3.16.4 Proiba Logins de root

Assim como é incomum trabalhar como usuário root, também é considerado inseguro permitir acesso root através da rede diretamente. Usuários remotos usualmente conectam com seu login próprio e depois dão um su para root quando necessário. Para se precaver de crackers, hackers e usuários ignorantes de entrarem diretamente como root, edito o arquivo /etc/securetty e comente (iniciando a linha com '#') tudo menos os terminais locais (console, tty1, tty2, etc).

Depois deste "conserto", usuários que tentar entrar como root pela rede receberão uma mensagem negando a parmissão:

```
Linux 2.0.29 (durak.interactive.com) (ttyp4)

durak login: root
root login refused on this terminal.
durak login:
```

14.3.16.5 Verifique se há patches disponívies

Como uma distribuição ativamente mantida, as atualizações e patches do Slackware estão disponíveis em:

http://www.slackware.com/changelog/

14.3.16.6 Mantenha-se informado

Você pode querer se inscrever em uma ou mais listas de e-mail que alertam usuários sobre falhas no Linux, como em:

http://www.slackware.com/lists/

http://www.cert.org/ (não só Linux, mas de várias falhas em sistemas de Internet)

14.3.17 Back up

Gostou de como as coisas estão rodando? Salve-as de uma tempestade mantendo cópias de segurança. O Amanda (Advanced Maryland Automatic Network Disk Archiver) é uma das muitas opções de backups disponíveis para Linux. Você pode aprender mais sobre o Amanda em:

http://www.amanda.org

15 ENFRENTANDO DIFICULDADES DE INSTALAÇÃO

Quase todo o mundo passa por algum tipo de dificuldade na primeira vez que tenta instalar o Linux. A maior parte das vezes, o problema é causado por simples mal-entendido. Outras vezes, porém, pode ser mais sério, como um equívoco do desenvolvedor ou um bug.

Se sua instalação parece ter sido bem sucedida, mas você recebe mensagens de erro inesperadas, veja a seguir.

15.1 Problemas inicializando através da mídia de instalação.

Quando tentar inicializar através da mídia de instalação pela primeira vez, você pode encontrar uma série de problemas. Estes são descritos a seguir. Note que esses problemas não estão relacionados à inicialização do novo sistema Linux instalado.

• Erro de disquete ou de mídia ao tentar inicializar.

A causa mais comum deste tipo de problema é um disquete corrompido. Ou o disco está fisicamente danificado - e neste caso você deve recriar o disquete com um novo -, ou os dados no disco estão mal gravados, caso no qual você deve verificar que tenha feito download e gravado corretamente a imagem. Em geral, simplesmente refazer o disquete vai solucionar o problema. Siga os passos novamente e tente outra vez.

Se você recebeu o disquete de uma empresa, contate essa empresa e peça um novo disquete - mas somente depois de estar certo que esse realmente é o problema.

O sistema pendura durante a inicialização ou após a mesma.

Depois que o sistema iniciar, você vai ver uma série de mensagens do kernel indicando quais dispositivos foram reconhecidos e configurados. Depois disso, normalmente aparece o prompt de login, permitindo que você continue a instalação (algumas distribuições jogam você diretamente em algum tipo de programa de instalação). O sistema pode parecer pendurar em muitos desses passos. Seja paciente: carregar software do disquete é um processo bastante lento. Em muitos casos, o sistema não pendurou, mas está meramente levando um bom tempo. Verifique se não há atividade de disco ou do sistema (mudanças na tela) por vários minutos antes de assumir que tenha pendurado.

- Depois de iniciar do prompt do LILO, o sistema deve carregar a imagem do kernel do disquete. Isso pode levar vários segundos. Você vai saber que ele está carregando pela luz ligada do drive.
- Enquanto o kernel inicializa, os dispositivos SCSI devem ser sondados. Se você não tem dispositivo SCSI instalado, o sistema vai pendurar por uns 15 segundos enquanto a sondagem SCSI continua. Isso ocorre depois de uma linha como esta aparecer na tela:

lp_init: lp1 exists (0), using pollin driver

3. Depois que o kernel terminar a inicialização, o controle é transferido para os arquivos de inicialização do sistema do disquete. Finalmente, será apresentado um prompt de login, ou vai aparecer um programa de instalação. Se aparecer um prompt de login, como:

Linux login:

você deve logar no sistema (usualmente como root ou "install" - isso varia conforme a distribuição). Depois de entrar o nome de usuário, o sistema pode dar uma pausa de 20 segundos ou mais enquanto o programa de instalação ou script é carregado do disquete. Novamente, a luz do drive deve ligar. Não pense que o sistema pendurou.

Quaisquer dos itens acima podem ser fonte de problemas. Entretanto, pode ser que o sistema realmente pendure durante a inicialização devido a diversos fatores. Você pode não ter memória suficiente para inicializar da mídia de instalação, por exemplo.

A causa de muitos "congelamentos" do sistema é incompatibilidade de hardware. O último capítulo apresenta uma visão geral sobre hardware suportado no Linux. Mesmo que seu hardware seja suportado, você pode passar por dificuldades com configurações incompatíveis de hardware, que podem fazer o sistema pendurar.

O sistema acusa falta de memória ao tentar inicializar ou instalar o software.

Provavelmente você vai enfrentar este problema se estiver instalando em um velho, velho, velho sistema com 4MB de RAM ou menos. Isso por que muitas distribuições usam um disco em RAM - um sistema de arquivos carregado diretamente na RAM para operações durante a instalação. A imagem inteira do disquete de instalação, por exemplo, pode ser carregada num disco em RAM, o que pode ocupar mais de 1MB.

Voce pode não ver uma mensagem "out of memory" quando tentar instalar ou inicializar; ao invés disso, o sistema pode pendurar inesperadamente ou falhar na inicialização. Se ele pendura, e você não encontra uma explicação (e tem pouca memória mesmo), tente desabilitar o ramdisk. Consulte a documentação da sua distribuição.

O Linux precisa de no mínimo 2 MB de RAM para rodar, mas algumas distribuições mais novas requerem bem mais do que isso.

O sistema informa um erro como "permission denied" (permissão negada) ou "file not found" (arquivo não encontrado)enquanto inicializa.

Essa é uma indicação que sua mídia de instalação pode estar corrompida. Se você tentar inicializar a partir da mídia de instalação (estando certo de estar fazendo tudo corretamente), você não deve ver erros desse tipo. Entre em contato com o distribuidor do seu Linux e verifique o problema - talvez tenha que conseguir novos discos de

inicialização. Se você mesmo fez o download dos disquetes, tente recriá-los e veja se isso resolve.

• O sistema informa um erro do tipo "VFS: Unable to mount root" quando inicializando.

Essa mensagem de erro significa que o sistema de arquivos root (que está na mídia de instalação), não pôde ser encontrado. Ou sua mídia está corrompida de alguma forma, ou você não está inicializando o sistema corretamente.

Por exemplo, muitas distribuições em CD requerem que você mantenha o CD no drive quando inicializando. Esteja certo que o drive de CD-ROM está ok e verifique se há atividade. Pode ser que o sistema não esteja localizando o drive de CD-ROM na inicialização.

15.2 Problemas de Hardware

O problema mais comum ao tentar instalar ou usar o Linux é uma incompatibilidade de hardware. Mesmo que todo seu hardware seja suportado pelo Linux, uma configuração mal feita ou conflito de hardware pode, algumas vezes, causar resultados estranhos - seus dispositivos podem não ser reconhecidos na inicialização ou o sistema pendurar.

É importante isolar esses problemas de hardware se você suspeitar que eles possam ser a fonte de seus problemas.

15.2.1 Isolando problemas de hardware

Se você enfrentar um problema que acredita estar relacionado ao hardware, a primeira coisa a tentar é isolar o problema. Isso significa eliminar todas as variáveis possíveis e (usualmente) desmontar o sistema, peça por peça, até que a peça de hardware seja isolada.

Isso não é tão assustador quanto possa parecer. Basicamente, você deve remover todo o hardware não essencial ao sistema, e então determinar qual dispositivo está causando o problema - possivelmente reinstalando cada dispositivo, um por vez. Isso significa que você deveria remover todo o hardware exceto os controladores de vídeo e disquete e, claro, o teclado. Mesmo dispositivos aparentemente inocentes, como os controladores de mouse, podem causar perturbação a sua paz de espírito a menos que você os considere não essenciais.

Por exemplo: vamos assumir que seu sistema pendure durante a detecção da placa de rede. Você pode criar uma hipótese de que há um conflito ou problema com a placa de rede de sua máquina. A maneira mais fácil de saber isso é retirar a placa de rede e tentar reinicializar. Se tudo der certo, você saberá que ou a placa não é suportada, ou há um conflito de IRQ ou endereço E/S com a placa.

"Conflito de IRQ ou endereço de E/S?" O que isso quer dizer? Todos os dispositivos na sua máquina usam uma IRQ (Interrupt Request Line, ou linha de interrupção) para se comunicar com o sistema e dizer que eles precisam

que algo seja feito. Você pode pensar numa IRQ como se fosse uma corda que o dispositivo puxa quando precisa que o sistema tome conta de alguma requisição. Se mais de um dispositivo estiver puxando na mesma corda, o kernel não poderá determinar qual dispositivo precisa de atenção.

Assim sendo, esteja certo que todos os dispositivos instalados usem IRQs únicas. Em geral, a IRQ para um dispositivo pode ser configurada por jumpers na placa; veja a documentação de cada dispositivo em particular. Alguns dispositivos não precisam de uma IRQ, mas é recomendado que você use uma se possível (os controladores SCSI Seagate ST01 e ST02 são bons exemplos).

Em alguns casos, o kernel fornecido na sua mídia de instalação é configurado para usar certas IRQs para certos dispositivos. Em algumas distribuições Linux, por exemplo, o kernel é pré-configurado para usar IRQ 5 para o controlador SCSI TMC-950, o CD-ROM Mitsumi, e o driver do bus mouse. Se você quer usar dois ou mais desses dispositivos, você precisa primeiro instalar o Linux com um deles habilitado, e então recompilar o kernel para mudar a IRQ padrão para os outros.

Outras áreas onde o hardware pode causar conflitos são no DMA (Direct Memory Access - Acesso Direto à Memória); endereços de E/S; e endereços de memória compartilhada. Todos esses termos descrevem mecanismos através dos quais o sistema se comunica com os dispositivos de hardware. Algumas placas de rede, por exemplo, usam endereço de memória compartilhada assim como uma IRQ para se comunicarem com o sistema. Se qualquer desses mecanismos conflitar com outros dispositivos, o sistema pode se comportar de forma imprevista. Você deve ser capaz de mudar o canal de DMA, endereço de E/S ou de memória compartilhada para vários dispositivos através configurações de jumpers. Desafortunadamente, alguns dispositivos não permitem mudança dessas configurações.

A documentação dos dispositivos de hardware deve especificar a IRQ, canal de DMA, endereço de E/S e endereço de memória compartilhada que esses dispositivos usam. Novamente, a maneira mais simples de contornar esses problemas é desabilitando temporariamente o dispositivo conflitante até que você tenha tempo para determinar a causa do problema.

A tabela abaixo é uma lista de IRQ e canais de DMA usados por vários dispositivos "padrão" na maioria dos sistemas. Quase todos os sistemas têm esses dispositivos. Assim, é melhor evitar configurar o IRQ ou DMA de outros dispositivos em conflito com esses valores.

Device	I/O address	IRQ	DMA
ttyS0 (COM1)	3f8	4	n/a
ttyS1 (COM2)	2f8	3	n/a
ttyS2 (COM3)	3e8	4	n/a
ttyS3 (COM3)	2e8	3	n/a
lp0 (LPT1)	378 - 37f	7	n/a
lp1 (LPT2)	278 - 27f	5	n/a
fd0, fd1 (floppies 1 and 2)	3fD - 3f7	6	2
fd2, fd3 (floppies 3 and 4)	37D - 377	10	3

Tabela 2.8: Configuração de alguns dispositivos comuns.

15.2.2 Problemas no reconhecimento de um disco rígido ou controlador

Quando o Linux inicializa, você deve ver uma série de mensagens na tela, como:

```
Console: colour EGA+ S0x25, S virtual consoles
Serial driver version 3.96 with no serial options enabled
tty00 at 0x03f8 (irq = 4) is a 16450
tty03 at 0x02e8 (irq = 3) is a 16550A
lp_init: lp1 exists (0), using polling driver
...
```

Aqui, o kernel está detectando vários dispositivos de hardware presentes no seu sistema. Depois, você vai ver uma linha como:

```
Partition check:
```

seguidas de uma lista de partições reconhecidas, como:

```
Partition check:
hda: hda1 hda2
hdb: hdb1 hdb2 hdb3
```

Se, por alguma razão, seus discos ou partições não são reconhecidos, você não poderá acessá-los.

Há muitas razões para isto:

- Controlador ou disco rígido não suportados. Se você tem um controlador de disco rígido (IDE, SCSI, ou outro) que não seja suportado pelo Linux, o kernel não vai reconhecer suas partições no momento da inicialização.
- Controlador ou disco configurado impropriamente. Mesmo que seu controlador seja suportado pelo Linux, pode não estar configurado corretamente. Este é um problema comum em controladores SCSI. A maior parte de controladores não SCSI devem funcionar bem sem nenhuma configuração adicional.

Refira-se à documentação de seu disco rígido e/ou controlador. Em particular, muitos discos rígidos precisam de uma configuração de jumpers para serem usados com "slave" (o segundo dispositivo tanto na IDE primária ou secundária). O teste decisivo nesta condição é inicializar no MS-DOS ou outro sistema operacional que você saiba que

funcione com seu disco ou controlador. Se puder acessar o disco e controlador de outro sistema operacional, não é problema de hardware.

- Controlador configurador corretamente, mas não reconhecido. Alguns controladores SCSI sem BIOS requerem que o usuário especifique informação sobre o controlador no momento de inicialização.
- Geometria do disco não reconhecida. Alguns sistemas, como o IBM PS/ValuePoint, não armazenam a informação de geometria do disco na memória CMOS, onde o Linux espera encontrá-la. Além disso, certos controladores SCSI precisam que você indique onde encontrar a geometria do disco para que o Linux reconheça a configuração do disco.

A maioria das distribuições fornece uma opção de especificar a geometria do disco. Geralmente, quando inicializar de uma mídia de instalação, você pode especificar a geometria no prompt do LILO, com um comando como:

boot: linux hd=cilindros,cabeças,setores

onde cilindros, cabeças e setores correspondem ao número de cilindros, cabeças e setores por trilha para o seu disco rígido.

Depois de instalar o Linux, você poderá instalar o LILO, permitindo que você inicialize do disco rígido. Você pode especificar a geometria do disco para LILO, tornando desnecessário entrar a geometria cada vez que você inicializa.

15.2.3 Problemas com dispositivos e controladores SCSI

Apresentamos aqui uma série de problemas comuns com controladores SCSI e dispositivos como CD-ROMs, discos rígidos e drives de fita. Se você tem problemas para fazer o Linux reconhecer seu disco ou controlador, leia adiante.

O HOWTO sobre SCSI contém muita informação útil sobre dispositivos SCSI além da listada aqui. O SCSI pode ser particularmente manhoso para configurar algumas vezes.

- Um dispositivo SCSI é detectado em todos os IDs possíveis. Isso é
 causado por configurar o dispositivo para o mesmo endereço do
 controlador. Você tem que mudar o jumper para que o disco use um
 endereço diferente do controlador.
- Linux reporta erros no acesso, mesmo que o dispositivo seja sabidamente livre de defeitos. Isso pode ser causado por cabos defeituosos ou terminadores ruins. Se o seu barramento SCSI não possuir terminadores nas duas extremidades, você pode enfrentar erros acessando o dispositivo. Quando em dúvida, verifique sempre os cabos.
- Dispositivos SCSI indicam erros de timeout. Isso é usualmente causado por conflitos de IRQ, DMA ou endereço de E/S. Também verifique se as interrupções estão corretamente habilitadas no controlador.
- Controladores SCSI que usam BIOS não são detectados. Detecção de controladores que usam BIOS irá falhar se a BIOS estiver

desabilitada, ou se a assinatura do seu controlador não é reconhecida pelo kernel. Veja o Linux SCSI HOWTO disponível em www.linuxdoc.org.

- Controladores que usam memória de E/S mapeada não funcionam.
 Isso é causado quando as portas de E/S mapeadas em memória estão incorretas no cache. Marque o espaço de endereço usado pela placa como "não cacheável" na configuração da CMOS, ou desabilite o cache totalmente.
- Quando particionando, você recebe um aviso "cylinders > 1024", ou não consegue inicializar de uma partição usando cilindros acima do 1023. A BIOS limita o número de cilindros a 1024, e qualquer partição usando cilindros acima desta posição não serão acessíveis através da BIOS. Quanto ao Linux, isso só afeta a inicialização; uma vez que o sistema tenha inicializado você deve estar apto a acessar a partição. Suas opções são: inicializar de um disquete ou de uma partição usando cilindros abaixo do 1024.
- Drive de CD-ROM ou outro dispositivo de mídia removível não são reconhecidos na inicialização. Tente reinicializar com um CD-ROM (ou disco) no drive. Isso é necessário para alguns dispositivos.

Se o seu controlador SCSI não é reconhecido, você pode tentar forçar a detecção de hardware no momento da inicialização. Isso é particularmente importante para controladores SCSI sem BIOS. A maioria das distribuições permite que você especifique a IRQ da placa controladora e o endereço de memória compartilhada quando estiver iniciando da mídia de instalação. Por exemplo, se estiver usando um controlador TMC-8xx, você pode entrar:

boot: linux tmxSxx=interrupção,endereço-de-memória

no prompt do LILO, sendo que interrupção é a IRQ da controladora, e endereço-de-memória é o endereço da memória compartilhada. Se isso é possível ou não, depende da distribuição Linux que estiver usando. Consulte sua documentação para detalhes.

15.3 Problemas instalando software

Instalar o Linux deve ser uma operação bem simples, se você tiver sorte. Os únicos problemas que você pode enfrentar seriam relacionados à mídia de instalação com defeito ou falta de espaço em disco. Aqui temos uma lista desses problemas comuns:

• O sistema reporta "Read error" (erro de leitura), "file not found" (arquivo não encontrado), ou outros errps enquanto tenta instalar o software. Isso indica um problema com a mídia de instalação. Se estiver instalando de disquetes, lembre-se que eles são bastante suscetíveis a erros deste tipo. Use disquetes novos, recém formatados. Se você tem uma partição MS-DOS, muitas distribuições permitem que instale o software a partir dela. Isso é mais rápido e confiável do que usar disquetes.

Se estiver usando um CD-ROM, veja se ele não está danificado,

com poeira, ou outra coisa que possa estar causando erros de leitura.

A causa do problema pode estar relacionada ao formato da mídia. Algumas distribuições requerem que os disquetes sejam formatados no formato MS-DOS de alta densidade. O disco de inicialização é exceção, (ele não é em formato MS-DOS na maioria dos casos). Se tudo o mais falhar, obtenha um novo conjunto de disquetes ou gere-os novamente (usando novos disquetes) se você mesmo fez download do software.

- O sistema reporta erros como "tar: read error" ou "gzip: not in gzip format". Esse problema é causado geralmente por arquivos corrompidos na mídia de instalação. Em outras palavras, seu disquete pode estar ok, mas os dados gravados no disquete estão, de alguma forma, corrompidos. Se você fez download do Linux usando modo texto (ASCII), ao invés de modo binário, então seus arquivos estarão corrompidos e não poderão ser lidos pelo software de instalação.
- O sistema reporta erros como "device full" durante a instalação. Este é um sinal bem claro de que acabou seu espaço em disco. Nem todas as distribuições conseguem lidar com este problema de forma limpa. Você provavelmente não vai conseguir abortar a instalação e conseguir que o sistema funcione.

A solução normalmente é recriar seu sistema de arquivos (com mke2fs), o que apaga os software parcialmente instalados. Você pode tentar reinstalar o software selecionando menos pacotes. Em outros casos, você pode ter que recomeçar completamente e repensar o tamanho de suas partições e sistemas de arquivos.

• O sistema reporte erros como "read_intr: 0x10" durante acesso ao disco rígido. Isso normalmente indica blocos defeituosos no disco. No entanto, se você receber esses erros enquanto estiver usando o mkswap ou mke2fs, o sistema pode estar tendo problemas para acessar seu disco. Isso pode ser um problema de hardware ou pode ser devido a uma geometria pobremente especificada. Se você usou a opção:

hd=cilindros, cabeças, setores

no momento da inicialização para forçar a geometria do disco e especificou a geometria incorretamente, isso pode ser a causa do problema. Isso também pode acontecer se a geometria do seu disco está mal especificada na CMOS.

• O sistema reporta erros como "file not found" ou "permission denied". Este problema pode ocorrer se nem todos os arquivos necessários estão presentes na mídia de instalação (veja próximo parágrafo) ou se há problemas de permissão com o software de instalação. Algumas distribuições Linux, por exemplo, têm bugs conhecidos no programa de instalação. Esses bugs normalmente são corrigidos rapidamente e são pouco freqüentes. Se você suspeita que sua distribuição tem bugs e está certo de não ter feito nada errado, entre em contato com o mantenedor da distribuição e reporte o bug.

Se você enfrentar outros erros estranhos ao instalar o Linux (especialmente se você fez download do Linux por conta própria), verifique se obteve todos os arquivos necessários. Por exemplo, algumas pessoas usam o

comando FTP:

mget *.*

ao fazer download do Linux via FTP. Isso apenas irá fazer download de arquivos que contenham um ponto (".") no nome do arquivo; se existirem arquivos sem ".", você não vai pegá-los. O comando correto nesse caso é

mget *

O melhor conselho é reconstruir seus passos quando algo der errado. Você pode pensar que fez tudo corretamente quando, de fato, esqueceu um pequeno, mas importante, passo em alguma parte do caminho. Em muitos casos, baixando e instalando novamente o software pode resolver o problema. Não bata a cabeça na parede mais do que precisa!

15.4 Problemas após instalar o Linux

Você gastou uma tarde inteira instalando o Linux. Para fazer espaço para ele, você apagou suas partições de MS-DOS ou OS/2 e chorando apagou suas cópias de SimCity e Wing Commander. Você reinicia o sistema. . . e nada acontece. Ou, ainda pior, algo acontece, mas não é o que deveria. O que fazer?

Além dos problemas já abordados, você pode ser vítima dos seguintes males:

15.4.1 Problemas ao inicializar o Linux do disquete

Se você usa um disquete para inicializar o Linux, você pode ter que especificar a localização da partição root no momento do boot. Isso é especialmente verdade se você está usando o disquete de instalação original, e não um disquete personalizado criado durante a instalação.

Enquanto inicializa do disquete, pressione Shift ou Ctrl. Com isso, um menu deve aparecer. Pressione Tab para ver uma lista de opções disponíveis. Por exemplo, muitas distribuições permitem que você digite:

boot: linux hd=partição

no menu de inicialização, onde partição é o nome da partição root do Linux (ex: /dev/hda2). Consulte a documentação de sua distribuição para detalhes.

15.4.2 Problemas inicializando o Linux de seu disco rígido

Se você optou por instalar o LILO ao invés de usar um disquete de inicialização, deve conseguir inicializar o Linux a partir do disco rígido. No entanto, o processo automático de instalação do LILO usado por muitas distribuições nem sempre é perfeito. Ele pode fazer suposições incorretas sobre o formato de sua partição e você terá que reinstalar o LILO para que tudo funcione corretamente.

• O sistema reporta "Drive not bootable--Please insert system

disk." (disco não inicializável - por favor insira um disco de sistema). O MBR (Master Boot Record) do disco rígido foi corrompido de alguma forma. Na maioria dos casos, isso é inofensivo, e tudo no disco deve estar intacto. Há muitas maneiras de contornar isto:

1. Ao particionar seu disco usando fdisk, você pode ter apagado a partição que estava marcada com ativa. O MS-DOS e outros sistemas operacionais tentam inicializar a partição ativa no boot (o Linux não se importa se a partição está "ativa" ou não). Você pode inicializar o MS-DOS de um disquete e rodar o FDISK.EXE para marcar sua partição MS-DOS como ativa e tudo ficará bem.

Outro comando para tentar (no MS-DOS 5.0 ou posterior) é "FDISK /MBR". Este comando tenta reconstruir o MBR de seu disco rígido para inicializar com o MS-DOS, sobreescrevendo o LILO. Se você não tem mais o MS-DOS no seu disco, você precisa inicializar o Linux de um disquete e tentar instalar o LILO depois.

- 2. Se você criou uma partição MS-DOS usando a versão Linux do fdisk, ou vice-versa, você pode chegar neste erro. Você deve criar partições MS-DOS usando a versão MS-DOS FDISK.EXE (Isso se aplica também a outros sistemas operacionais). A melhor solução é começar tudo novamente e reparticionar o disco corretamente, ou simplesmente apagar e recriar a partição com problema com a versão correta do fdisk.
- 3. O processo de instalação do LILO pode ter falhado. Neste caso, você pode inicializar do disquete de inicialização (se tiver um), ou da mídia de instalação original. Qualquer desses meios deve dar possibilidade de especificar a partição root a ser usada quando da inicialização. Aperte e segure Shift ou Ctrl durante o boot, e pressione Tab a partir do menu para uma lista de opções.
- Ao inicializar o sistema do disco rígido, o Windows (ou qualquer outro sistema operacional) é iniciado no lugar do Linux. Primeiramente, certifique-se que você realmente tenha instalado o LILO quando instalou o Linux. Se não, o sistema ainda vai iniciar o Windows (ou outro sistema que você tenha) quando tentar inicializar do disco rígido. Para dar boot do Linux a partir do disco rígido, você deve instalar o LILO.

Se você realmente instalou o LILO e outro sistema operacional ainda entra no lugar do Linux, então seu LILO está configurado para inicializar o outro sistema operacional por padrão. Enquanto o sistema estiver inicializando, segure pressionado Shift ou Ctrl, e pressione Tab durante o prompt do boot. Isso deve apresentar uma lista dos sistemas operacionais disponíveis para boot. Selecione a opção apropriada (usualmente "linux") para inicializar o Linux.

Se você quer que o Linux seja o seu sistema padrão, deve reconfigurar ou reinstalar o LILO.

Também é possível que você tenha tentado instalar o LILO, mas a instalação tenha falhado de alguma forma. Veja a seção anterior.

15.4.3 Problemas para entrar no sistema (login)

Depois de inicializar o Linux, provavelmente você receberá um prompt de login, como:

linux login:

Neste ponto, ou a documentação de sua distribuição ou o próprio sistema indicarão o que fazer. Em muitas distribuições, você simplesmente entra como "root", sem senha. Outros nomes possíveis são "guest" ou "test".

A maior parte dos sistemas Linux recém instalados não requerem uma senha para o login inicial. Entretanto, se o sistema pedir uma senha, pode ser um problema. Primeiro, tente usar uma senha igual ao nome de usuário, isto é, se você entrou como "root", use "root" como a senha. Lembre-se também se o programa de instalação não pediu que você cadastrasse uma senha. Se pediu, você deve tê-la anotado em algum lugar.

Se você simplesmente não consegue entrar, pode ser um problema. Primeiro, consulte a documentação de sua distribuição; o nome de usuário e senha que você deve usar deve estar gravado nela em algum lugar. Eles também podem ter sido fornecidos a você durante o processo de instalação, ou estar sendo mostrados na mensagem de login.

Uma causa pode ser um problema na instalação do programa de login do Linux e arquivos de inicialização. Você pode ter que reinstalar (ao menos partes) o Linux, ou reinicializar sua mídia de instalação para tentar resolver o problema manualmente.

15.4.4 Problemas usando o sistema

Se for bem sucedido ao entrar no sistema, você verá o prompt do shell (como "#", ou "\$") e pode passear alegremente por ele. No entanto, podem aparecer alguns problemas iniciais no uso do mesmo.

O problema de configuração mais comum é o de permissões incorretas de arquivos ou diretórios. Isso pode causar a mensagem de erro "shell-init: permission denied" após o login (de fato, em qualquer momento que você vir a mensagem "permission denied", você pode estar certo de que se trata de um problema de permissões).

Em muitos casos, basta simplesmente usar o chmod para corrigir as permissões dos arquivos ou diretórios apropriados. Algumas distribuições Linux, por exemplo, usaram uma vez (incorretamente) o mode 0644 para o diretório root (/). A solução para esta questão é executar o comando "chmod 755 /" como root. No entanto, para executar este comando, você precisava inicializar da mídia de instalação e montar seu sistema de arquivos manualmente - uma tarefa cabeluda para a maioria dos iniciantes.

Usando o sistema, você pode encontrar problemas com arquivos ou diretórios com permissões incorretas, ou softwares que não funcionam como deveriam. Bem vindo ao mundo Linux! Apesar da maioria das distribuições ser bastante livre de problemas, poucas delas são perfeitas. Não pretendemos cobrir todos os problemas aqui. Ao invés disso, ao longo do livro tentamos ajudá-lo a resolver muitos desses problemas de configuração mostrando como você mesmo pode solucioná-los.