Referência Debian

Osamu Aoki `<osamu\#at\#debian.org>`
Paulo Rogério Ormenese (líder: pt-br) `<pormenese\#at\#uol.com.br>`
'Autores' on page 257

CVS, Qui Jan 18 11:53:12 UTC 2007

Resumo

Esta Referência Debian (`http://qref.sourceforge.net/`) dedica-se a fornecer uma visão ampla do sistema Debian como um **guia do usuário pós-instalação**. Ela abrange muitos aspectos de administração do sistema através de exemplos de **comando shell**. São proporcionados tutoriais básicos, dicas, e outras informações para temas incluindo conceitos fundamentais do sistema Debian, sugestões de instalação de sistema, administração de pacotes Debian, o kernel Linux sob o Debian, ajustes do sistema, construção de um gateway, editores de texto, CVS, programação, e GnuPG para **não-desenvolvedores**.

Nota de Copyright

Sumário

1 Prefácio **1**

1.1 Documento Oficial 1

1.2 Convenções do documento 2

1.3 Scripts de exemplo 2

1.4 Configuração básica 2

1.5 Básico das distribuições Debian 3

2 Fundamentos Debian **5**

2.1 Os repositórios Debian 5

2.1.1 Estruturas de diretório 5

2.1.2 Distribuições Debian 6

2.1.3 A distribuição `stable` 6

2.1.4 A distribuição `testing` 7

2.1.5 A distribuição `unstable` 7

2.1.6 A distribuição `frozen` 8

2.1.7 Codinomes das distribuições Debian 8

2.1.8 Codinomes usados no passado 8

2.1.9 A origem dos codinomes 8

2.1.10 O diretório `pool` 9

2.1.11 Notas históricas sobre a Sid 9

2.1.12 Pacotes enviados para `incoming/` 10

2.1.13 Obtendo um pacote antigo 10

2.1.14 Seções de Arquiteturas 10

2.1.15 O código-fonte 11

2.2 O sistema de gerenciamento de pacotes Debian 11
  2.2.1 Visão Geral dos pacotes Debian 11
  2.2.2 Formato de pacotes Debian 12
  2.2.3 Convenções de nomenclatura para nomes de arquivos de pacotes Debian 13
  2.2.4 Preservação da configuração local 13
  2.2.5 Scripts de manutenção Debian 14
  2.2.6 Prioridade de pacotes 14
  2.2.7 Pacotes virtuais 15
  2.2.8 Dependências de pacotes 15
  2.2.9 O significado de Pré-dependências 16
  2.2.10 Estado do pacote 17
  2.2.11 Evitando que pacotes sejam atualizados 17
  2.2.12 Pacotes fonte 18
  2.2.13 Construindo pacotes binários a partir de um pacote fonte 18
  2.2.14 Criando novos pacotes Debian 19
2.3 Atualizando um sistema Debian 19
  2.3.1 Métodos para atualizar um sistema Debian 19
  2.3.2 Visão geral das ferramentas de gerenciamento de pacotes 19
  2.3.3 `dpkg` 20
  2.3.4 APT 20
  2.3.5 `dselect` 20
  2.3.6 Atualizando um sistema em execução 21
  2.3.7 Arquivos `.deb` baixados e em cache 21
  2.3.8 Mantendo registros para atualizações 21
2.4 O processo de inicialização Debian 21
  2.4.1 O programa `init` 21
  2.4.2 Níveis de execução 22
  2.4.3 Personalizando o processo de inicialização 22
2.5 Suportando diversidades 23
2.6 Internacionalização 24
2.7 Debian e o kernel 24

2.7.1 Compilando um kernel a partir de um fonte não Debian 24
2.7.2 Ferramentas para construir kernels personalizados 24
2.7.3 Condições especiais para lidar com módulos 25
2.7.4 Desinstalação de um kernel antigo 25

3 Dicas para instalação do Sistema Debian 27

3.1 Dicas para instalação de sistemas Linux em geral 27
3.1.1 O básico sobre a compatibilidade de hardware 27
3.1.2 Determinando hardware e chip set de um PC 28
3.1.3 Determinando o hardware de um PC via Debian 28
3.1.4 Determinando o hardware de um PC por meio de outros SO's 29
3.1.5 Um mito sobre o Lilo 29
3.1.6 GRUB 30
3.1.7 Escolha dos disquetes de inicialização 30
3.1.8 Instalação 30
3.1.9 Hosts e IP a usar para LAN 31
3.1.10 Contas de usuário 32
3.1.11 Criando sistemas de arquivo 32
3.1.12 Guia sobre memória DRAM 35
3.1.13 Espaço de troca (swap) 36
3.2 Configuração do bash 36
3.3 Configuração do mouse 36
3.3.1 Mouses PS/2 36
3.3.2 Mouses USB 39
3.3.3 Touch pad 39
3.4 Configuração de NFS 40
3.5 Configuração do Samba 40
3.6 Configuração da Impressora 41
3.6.1 `lpr/lpd` 41
3.6.2 CUPS™ 42
3.7 Outras dicas de instalação no host 43

3.7.1 Instalar mais alguns pacotes após a instalação inicial 43
3.7.2 Módulos 43
3.7.3 Configuração básica de CD-RW 44
3.7.4 Memória grande e autodesligamento 44
3.7.5 Estranhos problemas de acesso a alguns websites 45
3.7.6 Configuração Dialup PPP 45
3.7.7 Outros arquivos de configuração para fuçar em `/etc/` 46

4 Tutoriais do Debian 47

4.1 Começando 47
4.1.1 Login em um prompt shell como root 47
4.1.2 Configurar um ambiente mínimo para novato 48
4.1.3 Adicionar uma conta de usuário 49
4.1.4 Alternar entre consoles virtuais 49
4.1.5 Como desligar 49
4.1.6 Hora do jogo 50
4.2 Midnight Commander (MC) 50
4.2.1 Melhorar o MC 51
4.2.2 Iniciar o MC 51
4.2.3 Gerenciador de arquivos no MC 51
4.2.4 Truques de linha de comando no MC 52
4.2.5 Editor no MC 52
4.2.6 Visualizador no MC 53
4.2.7 Recursos de auto início no MC 53
4.2.8 Sistema de arquivo virtual FTP no MC 53
4.3 Ambiente de trabalho semelhante ao Unix 53
4.3.1 Comandos de teclado especiais 54
4.3.2 Comandos Unix básicos 54
4.3.3 A execução de comando 58
4.3.4 Comando simples 58
4.3.5 Execução de comando e variável de ambiente 58

- 4.3.6 Caminho de busca de comando 59
- 4.3.7 Opções de linha de comando 59
- 4.3.8 Curingas (wildcards) de shell 60
- 4.3.9 Valor de retorno do comando 60
- 4.3.10 Seqüências de comando típicas 61
- 4.3.11 Apelido (alias) de comando 62

4.4 Processamento de texto em similares a Unix 63

- 4.4.1 Expressões regulares 63
- 4.4.2 Expressões de substituição 64

4.5 Sistema de arquivos em similares ao Unix 65

- 4.5.1 Básico de arquivos Unix 65
- 4.5.2 O conceito de sistema de arquivos no Debian 67
- 4.5.3 Permissões de acesso de arquivos e diretórios 67
- 4.5.4 Marcas de tempo 70
- 4.5.5 Ligações (links) 70
- 4.5.6 Pipes nomeados (FIFOs) 71
- 4.5.7 Soquetes 72
- 4.5.8 Arquivos de dispositivos 72
- 4.5.9 Sistema de arquivo `/proc` 73

4.6 Sistema X Window 73

- 4.6.1 Iniciar o Sistema X Window 73
- 4.6.2 Menu no Sistema X Window 74
- 4.6.3 Seqüência de teclado para o Sistema X Window 74

4.7 Estudos adicionais 74

5 Atualizando uma distribuição para a `testing` 77

5.1 Transição do APT para a versão Woody 77

5.2 Preparação para a transição (`stable` para `testing`) 78

5.3 Atualizar o sistema Debian 79

- 5.3.1 Melhor prática de atualização usando o `dselect` 79
- 5.3.2 Prática ruim de atualização com o `apt-get` 80

6 Gerenciamento de pacotes Debian 81

6.1 Introdução . . . 81
- 6.1.1 Ferramentas principais . . . 81
- 6.1.2 Ferramentas úteis . . . 82

6.2 Básico do gerenciamento de pacotes Debian . . . 82
- 6.2.1 Instalar *tarefa* com o `tasksel` ou `aptitude` . . . 82
- 6.2.2 Configure o sistema APT . . . 83
- 6.2.3 `dselect` . . . 83
- 6.2.4 `aptitude` . . . 84
- 6.2.5 Comandos `apt-cache` e `apt-get` . . . 84
- 6.2.6 Seguindo um tipo da distribuição Debian . . . 86
- 6.2.7 Rebaixando todos os pacotes para a `estável` . . . 86
- 6.2.8 Visão geral do arquivo `/etc/apt/preferences` . . . 87

6.3 Comandos de sobrevivência do Debian . . . 87
- 6.3.1 Verificar bugs no Debian e procurar ajuda . . . 88
- 6.3.2 Resolvendo problemas com atualização pelo APT . . . 88
- 6.3.3 Resgate usando o `dpkg` . . . 89
- 6.3.4 Recuperação dos dados de seleção de pacotes . . . 90
- 6.3.5 Resgate do sistema após quebrar `/var` . . . 90
- 6.3.6 Instalar um pacote em um sistema não iniciável . . . 90
- 6.3.7 O que fazer se o comando `dpkg` estiver quebrado . . . 91

6.4 Comandos de nirvana do Debian . . . 91
- 6.4.1 Informação sobre um arquivo . . . 91
- 6.4.2 Informação sobre um pacote . . . 92
- 6.4.3 Instalação com APT sem acompanhamento . . . 93
- 6.4.4 Reconfigurar pacotes instalados . . . 93
- 6.4.5 Remover e eliminar (purge) pacotes . . . 94
- 6.4.6 Mantendo pacotes antigos . . . 94
- 6.4.7 Sistema híbrido `stable/testing/unstable` . . . 94
- 6.4.8 Descartar arquivos de pacote em cache . . . 95
- 6.4.9 Gravar/copiar a configuração do sistema . . . 95

6.4.10 Portar um pacote para o sistema `stable` . . . 95
6.4.11 Repositório local de pacotes . . . 96
6.4.12 Converter ou instalar um pacote binário alien . . . 97
6.4.13 Verificar arquivos de pacotes instalados . . . 97
6.4.14 Otimizando o arquivo `sources.list` . . . 97
6.5 Outras peculiaridades do Debian . . . 98
6.5.1 O comando `dpkg-divert` . . . 98
6.5.2 O pacote `equivs` . . . 98
6.5.3 Comandos alternativos . . . 98
6.5.4 `init` System-V e níveis de execução (runlevels) . . . 99
6.5.5 Serviços daemon desabilitados . . . 100

7 O kernel Linux no Debian 101
7.1 Recompilação do Kernel . . . 101
7.1.1 Método Debian padrão . . . 101
7.1.2 Método clássico . . . 102
7.1.3 Cabeçalhos do kernel . . . 103
7.2 O kernel 2.4 modularizado . . . 103
7.2.1 PCMCIA . . . 104
7.2.2 SCSI . . . 104
7.2.3 Função de rede . . . 105
7.2.4 Sistema de arquivos EXT3 (> 2.4.17) . . . 106
7.2.5 Suporte Realtek RTL-8139 no kernel 2.4 . . . 107
7.2.6 Suporte a porta paralela . . . 107
7.3 Ajustando o kernel pelo sistema de arquivos proc . . . 107
7.3.1 Muitos arquivos abertos . . . 107
7.3.2 Intervalos de sincronização (flush) de disco . . . 108
7.3.3 Máquinas de pouca memória antigas e lerdas . . . 108

8 Dicas Debian 109
8.1 Iniciando o sistema . . . 109
8.1.1 "Eu esqueci a senha do root ¡‘ (1) . . . 109

8.1.2 "Eu esqueci a senha do root ¡‘ (2) . . . 110
8.1.3 Não posso iniciar o sistema . . . 110
8.1.4 "Deixe-me desabilitar o X na inicialização ¡‘ . . . 111
8.1.5 Outros truques de inicialização com o prompt de inicialização . . . 112
8.1.6 Definindo parâmetros de inicialização do GRUB . . . 112
8.2 Gravando atividades . . . 113
8.2.1 Gravando atividades shell . . . 113
8.2.2 Gravando atividades no X . . . 113
8.3 Copiar e arquivar um subdiretório inteiro . . . 114
8.3.1 Comandos básicos para copiar um subdiretório inteiro . . . 114
8.3.2 `cp` . . . 114
8.3.3 `tar` . . . 114
8.3.4 `pax` . . . 115
8.3.5 `cpio` . . . 115
8.3.6 `afio` . . . 115
8.4 Backups diferenciais e sincronização de dados . . . 116
8.4.1 Backup diferencial com rdiff . . . 116
8.4.2 Backup diário com `pdumpfs` . . . 117
8.4.3 Backup diferencial regular com RCS . . . 117
8.5 Recuperação de sistema paralizado . . . 117
8.5.1 Matar um processo . . . 117
8.5.2 Alt-SysRq . . . 118
8.6 Belos pequenos comandos para se lembrar . . . 118
8.6.1 Paginador . . . 118
8.6.2 Memória livre . . . 118
8.6.3 Definir a hora (BIOS) . . . 119
8.6.4 Definir a hora (NTP) . . . 119
8.6.5 Como controlar recursos de console como o protetor de tela . . . 119
8.6.6 Procurar na base de dados administrativa . . . 120
8.6.7 Desabilitar o som (beep) . . . 120
8.6.8 Mensagens de erro na tela do console . . . 120

8.6.9 Definir o console para o tipo correto 121
8.6.10 Voltar o console ao estador normal 121
8.6.11 Converter um arquivo texto DOS para o estilo Unix 121
8.6.12 Substituição de expressões regulares 122
8.6.13 Editar arquivo no local usando um script 122
8.6.14 Extrair diferenças e juntar atualizações para os arquivos fonte 122
8.6.15 Converter um arquivo grande em arquivos pequenos 123
8.6.16 Extrair dados de uma tabela em arquivo texto 123
8.6.17 Pequenos scripts para ligar comandos 124
8.6.18 Pequenos scripts para fazer loop para cada arquivo 125
8.6.19 Loucura de script Perl curto 126
8.6.20 Obter texto ou arquivo de lista de discussão de uma página web 126
8.6.21 Imprimir de forma bonita uma página web 126
8.6.22 Imprimir de forma bonita uma página de manual 127
8.6.23 Juntar dois arquivos PostScript ou PDF 127
8.6.24 Contar o tempo de um comando 127
8.6.25 Comando `nice` 127
8.6.26 Agendar atividade (`cron`, `at`) 128
8.6.27 Mudança de console com `screen` 128
8.6.28 Testes de rede básicos 130
8.6.29 Fazer flush em mensagens do spool local 130
8.6.30 Remover mensagens congeladas do spool local 130
8.6.31 Reentregar conteúdo mbox 130
8.6.32 Limpar conteúdo de arquivo 131
8.6.33 Arquivos Falsos 131
8.6.34 `chroot` 131
8.6.35 Como checar ligações diretas (hard) 133
8.6.36 Usar `mount` em um arquivo de uma imagem de disco rígido 134
8.6.37 Samba 134
8.6.38 Utilitários para sistemas de arquivos não nativos 134
8.7 Erros típicos a serem notados 134
8.7.1 `rm -rf .*` 135
8.7.2 `rm /etc/passwd` 135

9 Ajustando um sistema Debian 137
9.1 Inicialização do sistema 137
9.1.1 Personalizando os scripts init 137
9.1.2 Personalizando o log do sistema 138
9.1.3 Otimizando o hardware 138
9.2 Restringindo o acesso 139
9.2.1 Restringindo logins com PAM 139
9.2.2 ”Por que o GNU `su` não suporta o grupo `wheel`“ 140
9.2.3 Propósitos de grupos padrões 140
9.2.4 Trabalhando de forma mais segura – `sudo` 141
9.2.5 Restringindo acesso a serviços 142
9.2.6 Centralizando a autenticação – LDAP 142
9.3 Gravadores de CD 143
9.3.1 Introdução 143
9.3.2 Alternativa 1: módulos + `lilo` 143
9.3.3 Alternativa 2: recompilar o kernel 144
9.3.4 Passos posteriores à configuração 144
9.3.5 Arquivo imagem de CD (inicializável) 145
9.3.6 Gravar CDs (R, RW): 145
9.3.7 Fazer um arquivo imagem de um CD 146
9.3.8 Imagens de CD do Debian 146
9.3.9 Fazer cópia de segurança do sistema em CD-R 147
9.3.10 Copiar um CD de músicas para CD-R 147
9.4 X 147
9.4.1 Pacotes do X 148
9.4.2 Detecção de hardware para o X 149
9.4.3 O servidor X 149
9.4.4 Clientes X 151
9.4.5 Sessões X 153
9.4.6 Usando o X sobre TCP/IP 156
9.4.7 Conectando a um servidor X remoto – `xhost` 156

9.4.8 Conectando a um servidor X remoto – `ssh` 157
9.4.9 O emulador de terminal X – `xterm` 157
9.4.10 Recursos do X 157
9.4.11 Mapas de teclado e funções de botões do apontador no X 158
9.4.12 Obtendo root no X 158
9.4.13 Fontes TrueType no X 160
9.4.14 Navegadores Web no X 162
9.5 SSH 163
9.5.1 Básico do SSH 163
9.5.2 Redirecionamento de porta para tunelamento SMTP/POP3 165
9.5.3 Conectando com menos senhas – RSA 165
9.5.4 Lidando com clientes SSH estranhos 166
9.5.5 Configurando o `ssh-agent` 166
9.5.6 Resolução de problemas de SSH 167
9.6 Correio 167
9.6.1 Agentes de transporte de correio (MTAs) 167
9.6.2 Baixando mensagens – Fetchmail 170
9.6.3 Processando mensagens – Procmail 170
9.6.4 Lendo mensagens – Mutt 170
9.7 Localização 171
9.7.1 Básico da localização 171
9.7.2 Locales 172
9.7.3 Introdução a locales 172
9.7.4 Ativando suporte a locale 174
9.7.5 Ativando um locale específico 174
9.7.6 Locale de formato de data ISO 8601 175
9.7.7 Exemplo para os Estados Unidos (ISO-8859-1) 175
9.7.8 Exemplo para a França com o sinal de Euro (ISO-8859-15) 175
9.7.9 Exemplo para um sistema X window multi-idiomas 176
9.7.10 Métodos de entrada X alternativos 179
9.7.11 Emuladores de terminal no X 179

9.7.12 Exemplo para UTF-8 no X 180
9.7.13 Exemplo para UTF-8 em um console framebuffer 180
9.7.14 Além dos locales 180

10 Configuração de rede 183

10.1 Básico de rede IP 183
10.2 Configuração de rede de baixo nível 185
10.2.1 Configuração de rede de baixo nível – `ifconfig` e `route` 185
10.2.2 Configuração de rede de baixo nível – `ip` 187
10.2.3 Configurando uma interface Wi-Fi 187
10.2.4 Configurando uma interface PPP 188
10.3 Nomeando o computador 191
10.3.1 Nome do sistema (Hostname) 191
10.3.2 Nome de correio (Mailname) 192
10.4 Domain Name Service (DNS) (Serviço de Nome de Domínio) 192
10.4.1 O resolvedor (resolver) 192
10.4.2 Gerenciando informação do resolvedor de nomes – `resolvconf` 193
10.4.3 Colocando em cache nomes consultados – `dnsmasq` 193
10.4.4 Fornecendo Serviço de Nome de Domínio – `bind` 194
10.5 Configurando interfaces de rede usando DHCP 194
10.6 Configuração de rede de alto nível no Debian 195
10.6.1 Configurando uma interface com um endereço IP estático 195
10.6.2 Configurando uma interface usando DHCP 196
10.6.3 Configurando uma interface Wi-Fi 196
10.6.4 Configurando uma interface PPP 196
10.6.5 Configurando uma interface PPPoE 197
10.6.6 Configurando múltiplas interfaces Ethernet para um gateway 197
10.6.7 Configurando interfaces virtuais 198
10.7 **Re**configuração de rede 198
10.8 Reconfiguração mágica de rede 199
10.8.1 guessnet 201

10.8.2 laptop-net 201
10.9 Gatilhando a reconfiguração de rede 201
10.9.1 Gatilhando a configuração de rede durante a inicialização 202
10.9.2 Gatilhando configuração de rede – `hotplug` 202
10.9.3 Gatilhando a configuração de rede – `ifplugd` 203
10.9.4 Gatilhando a configuração de rede – `waproamd` 204
10.9.5 Configuração de rede e PCMCIA 204
10.10 Configuração de serviço de rede 205
10.11 Resolução de problemas com rede 206
10.12 Construindo um roteador gateway 207
10.12.1 Configuração do Netfilter 207
10.12.2 Gerenciar múltiplas conexões de rede 210

11 Editores 211

11.1 Editores populares 211
11.2 Editores de recuperação 212
11.3 Emacs e Vim 212
11.3.1 Dicas para o Vim 212
11.3.2 Dicas para o Emacs 212
11.3.3 Iniciando o editor 213
11.3.4 Resumo de comandos do Editor (Emacs, Vim) 213
11.3.5 Configuração do Vim 215
11.3.6 Ctags 215
11.3.7 Converter uma tela com sintaxe destacada em fonte HTML 216
11.3.8 Dividir a tela com o `vim` 216

12 Sistemas de Controle de Versão 217

12.1 Concurrent Versions System (CVS) (Sistema de Versões Concorrentes) 217
12.1.1 Instalando um servidor CVS 217
12.1.2 Exemplos de sessão CVS 218
12.1.3 Resolvendo problemas do CVS 221
12.1.4 Comandos CVS 221

12.2 Subversion 222

12.2.1 Instalando um servidor Subversion 222

12.2.2 Movendo um repositório CVS para Subversion 223

12.2.3 Exemplos de uso do Subversion 223

13 Programação 225

13.1 Onde começar 225

13.2 Shell 225

13.2.1 Bash – o shell interativo padrão do **GNU** 225

13.2.2 Shells POSIX 226

13.2.3 Parâmetros shell 227

13.2.4 Redirecionamento shell 227

13.2.5 Condicionais de shell 228

13.2.6 Processamento de linha de comando 229

13.3 Awk 230

13.4 Perl 231

13.5 Python 232

13.6 Make 233

13.7 C 234

13.7.1 Programa C simples (`gcc`) 235

13.7.2 Depurando 236

13.7.3 Flex – um Lex melhor 237

13.7.4 Bison – um Yacc melhor 237

13.7.5 Autoconf 238

13.8 Preparação de documentos 239

13.8.1 Processamento de texto `roff` 239

13.8.2 SGML 239

13.8.3 TeX/LaTeX 241

13.8.4 Programação Letrada (Literate Programming) 241

13.9 Empacotamento 243

13.9.1 Empacotando um binário simples 243

13.9.2 Empacotando com ferramentas 243

14 GnuPG **245**
14.1 Instalando GnuPG 245
14.2 Usando GnuPG 246
14.3 Gerenciando GnuPG 246
14.4 Usando GnuPG com aplicações 247
    14.4.1 Usando GnuPG com Mutt 247
    14.4.2 Usando GnuPG com o Vim 247

15 Suporte para o Debian **249**
15.1 Referências 249
15.2 Descobrindo o significado de uma palavra 253
15.3 Descobrindo a popularidade de um pacote Debian 253
15.4 O sistema de gerenciamento de bugs Debian 253
15.5 Listas de discussão 254
15.6 Internet Relay Chat (IRC) 254
15.7 Mecanismos de busca 255
15.8 Sítios Web 255

A Apêndice **257**
A.1 Autores 257
A.2 Garantias 260
A.3 Retorno 260
A.4 Formato do documento 260
A.5 O labirinto do Debian 261
A.6 As citações do Debian 261

Capítulo 1

Prefácio

Esta Referência Debian (`http://qref.sourceforge.net/`) tem como objetivo oferecer uma ampla visão do sistema Debian como um **guia do usuário pós-instalação**. O leitor alvo é alguém que esteja disposto a ler scripts shell. Espero que o leitor tenha conhecimentos anteriores em sistemas Unix-like antes de ler este documento.

Tomei uma decisão consciente de **não** explicar tudo em detalhes caso tais detalhes possam ser encontrados em uma **página de manual**, uma **página info** ou um **documento HOWTO**. Ao invés de explicações completas, eu tentei oferecer informação prática mais diretamente fornecendo as **sequências de comandos exatas** no texto principal ou **scripts de exemplo** em `http://www.debian.org/doc/manuals/debian-reference/examples/` como uma referência. Você deve entender o conteúdo dos exemplos antes de executá-los. Seu sistema pode requerer sequências de comandos ligeiramente diferentes.

Muita informação incluída consiste de lembranças ou apontadores para as referências oficiais listadas em 'Referências' on page 249. Isto é devido parcialmente a este documento ter sido originado como uma "**referência rápida**".

Matenha-o pequeno e simples (KISS) é meu princípio guia.

Para ajuda com manutenção de emergência do sistema, pule imediamente para o 'Comandos de sobrevivência do Debian' on page 87.

1.1 Documento Oficial

O último documento oficial está nos repositórios Debian com o nome de pacote `debian-reference-pt-br` e está também disponível em `http://www.debian.org/doc/manuals/debian-reference/`.

A última versão de desenvolvimento é `http://qref.sourceforge.net/Debian/`. O projeto é hospedado em `http://qref.sourceforge.net/`, onde este documento está disponível para download nos formatos texto puro, HTML, PDF, SGML e PostScript.

1.2 Convenções do documento

Este documento "Referência Debian" fornece informações através de pequenos comandos do shell Bash. Aqui estão algumas convenções usadas :

```
# comando na conta root
$ comando na conta do usuário
... descrição de ação
```

Consulte 'Bash – o shell interativo padrão do **GNU**' on page 225 para maiores informações sobre Bash.

Referência para:
- uma **página de manual Unix** é indicada na forma `bash(1)`.
- uma **página GNU TEXINFO** é indicado na forma `info libc`.
- um **livro** é indicado na forma *A linguagem de Programação C*.
- uma **URL** é indicada na forma `http://www.debian.org/doc/manuals/debian-reference/`.
- um **arquivo** no sistema é indicado na forma `/usr/share/doc/Debian/reference/`.

As seguintes abreviações são usadas :
- **LDP** : Projeto de Documentação Linux (`http://www.tldp.org/`)
- **DDP** : Projeto de Documentação Debian (`http://www.debian.org/doc/`)

Neste documento somente URLs são exibidas para documentos LDP, mas estes podem também ser obtidos como um pacote e instalados em `/usr/share/doc/HOWTO/`. Consulte 'Referências' on page 249.

1.3 Scripts de exemplo

Scripts de exemplo estão disponíveis no subdiretório de exemplos (`http://www.debian.org/doc/manuals/debian-reference/examples/`) instalado em `/usr/share/doc/Debian/reference/`; para arquivos ocultos, o "." inicial no nome do arquivo é convertido para caixa baixa "_". Uma extensão extra foi adicionada quando existem vários scripts de exemplo alternativos.

1.4 Configuração básica

Caso o sistema esteja instalado com o mínimo de pacotes, certifique-se de executar os seguintes comandos para instalar alguns pacotes essenciais e alguns documentos chave :

```
# aptitude install mc less ssh vim kernel-package \
                   manpages-dev doc-debian doc-linux-text \
```

```
debian-policy developers-reference maint-guide \
apt-howto harden-doc debian-reference \
libpam-doc glibc-doc samba-doc exim4-doc-html
```

1.5 Básico das distribuições Debian

O Debian é oferecido em 3 "sabores" de distribuições :

- `stable`: Boa para usar em um servidor de produção. Chato para uma estação de trabalho (WS). Consulte 'A distribuição `stable`' on page 6.
- `testing`: Boa para usar em uma WS. Consulte 'A distribuição `testing`' on page 7.
- `unstable`: Nunca use-a cegamente. Consulte 'A distribuição `unstable`' on page 7.

Leia pelo menos a lista de discussão chave `debian-devel-anounce@lists.debian.org` para atualizações sobre o estado do Debian.

Em Março de 2002, estas três versões de distribuições correspondiam às versões de codinomes Potato (qualidade de produção), Woody (beta-teste, já bastante estável quando o lançamento era iminente), e Sid (alfa-teste). Em Agosto de 2002, logo após o lançamento do Woody, as distribuições correspondiam a Woody (qualidade de produção), Sarge (beta-teste, teria algumas arestas ruins por um tempo) e Sid (sempre alfa-teste). Quando os pacotes na `unstable` não possuem bugs críticos ao lançamento (RC) cadastrados para eles após a primeira semana ou mais, eles são automaticamente promovidos para a `testing`. Consulte 'Os repositórios Debian' on page 5;

Há duas maneiras de habilitá-lo a executar as últimas versões de software em um sistema.
- 'Atualizando uma distribuição para a `testing`' on page 77 (principalmente para própositos de WS)
- 'Portar um pacote para o sistema `stable`' on page 95 (principalmente para prósitos de servidor)

Seguir o sabor `testing` do Debian tem o efeito colateral de obter correções de segurança muito lentamente. Então fique alerta.

Se você misturar sabores do Debian, usando partes de `testing` ou `unstable` na `stable`, você comumente será atualizado em muitos pacotes chaves de forma não intencional a partir da `testing` ou `unstable` que podem ter bugs. Então fique alerta.

Executar os sabores `testing` ou `unstable` do Debian envolve um risco aumentado de encontrar bugs sérios. Esse risco pode ser gerenciado implementando um esquema de multi-inicialização com um sabor mais estável do Debian, ou implementando o truque legal de usar o `chroot` com o mais estável como descrito em '`chroot`' on page 131. Esse último possibilitará executar diferentes sabores do Debian simultaneamente em consoles diferentes.

Após explicar alguns fundamentos sobre a distribuição Debian em 'Fundamentos Debian' on page 5, apresentarei alguma informação básica para ajudá-lo a viver alegremente com o último software, tirando vantagem das distribuições `testing` e `unstable` do Debian. Os impacientes devem pular imediatamente para 'Comandos de sobrevivência do Debian' on page 87. Feliz atualização !

Capítulo 2

Fundamentos Debian

Este capítulo fornece informação fundamental sobre o sistema Debian para não-desenvolvedores. Para informação oficial, consulte :
- Manual de Políticas Debian
- Referência do Desenvolvedor Debian
- Guia do Novo Desenvolvedor Debian

listados em 'Referências' on page 249.

Caso você esteja procurando por explicações "como-fazer" menos detalhadas, pule diretamente para 'Gerenciamento de pacotes Debian' on page 81 ou para outros capítulos relevantes.

Este capítulo é baseado em documentos obtidos da "FAQ Debian", altamente reorganizados para permitir que o administrador de sistemas Debian comum possa começar.

2.1 Os repositórios Debian

2.1.1 Estruturas de diretório

O software que foi empacotado para o Debian está disponível em uma das diversas árvores de diretórios em cada site espelho Debian (`http://www.debian.org/mirror/`) através de FTP ou HTTP.

Os seguintes diretórios podem ser encontrados em cada site espelho Debian sob o diretório `debian`:

`dists/`: Este diretório contém as "distribuições" e era a maneira canônica de acessar pacotes atuais disponíveis nas distribuições e pré-distribuições Debian. Alguns pacotes antigos, arquivos `Contents-*.gz`, e arquivos `Packages.gz` ainda estão aqui.

`pool/`: A nova localização física para todos os pacotes das distribuições e pré-distribuições Debian.

`tools/`: Utilitários DOS para criação de discos de partida, particionamento de seu disco, compactação/descompactação de arquivos e inicialização do Linux.

`doc/`: A documentação básica do Debian, como o FAQ, instruções do sistema de relatos de bugs, etc.

`indices/`: O arquivo Maintainers e os arquivos override.

`project/`: em sua maioria materiais somente para desenvolvedores, como :

> **`project/experimental/`:** Este diretório contém pacotes e ferramentas que ainda estão sendo desenvolvidos, e ainda estão em estágio alpha. Os usuários não deveriam estar usando pacotes daqui porque os mesmos podem ser perigosos e causarem danos mesmo para os mais experientes.
>
> **`project/orphaned/`:** Pacotes que foram abandonados por seus antigos mantenedores e retirados da distribuição.

2.1.2 Distribuições Debian

Nomalmente existem três distribuições Debian no diretório `dists`. Elas são nomeadas a distribuição `stable`, a distribuição `testing` e a distribuição `unstable`. Algumas vezes existe também uma distribuição `frozen`. Cada distribuição é definida como uma ligação simbólica para a diretório real com um codinome no diretório `dists`.

2.1.3 A distribuição `stable`

As entradas de pacotes para a distribuição `stable`, Debian Sarge (3.1r0), são mantidas no diretório `stable` (ligação simbólica para `sarge/`):

- `stable/main/`: Este diretório contém os pacotes que formalmente constituem a versão lançada mais recente do sistema Debian.

  Estes pacotes todos seguem as Linhas Guias Debian para o Software Livre (DFSG) (`http://www.debian.org/social_contract#guidelines`) (também disponível como `/usr/share/doc/debian/social-contract.txt` instalado por `debian-doc`) e são todos livremente utilizáveis e livremente distribuídos.

- `stable/non-free/`: Este diretório contém pacotes da distribuição que são restritos de alguma forma que requer que os distribuidores tenham cuidado especial com requerimentos de direitos de cópia especificados.

  Por exemplo, alguns pacotes possuem licenças que proíbem sua distribuição comercialmente. Outros podem ser redistribuídos mas são de fato shareware e não software livre. As licenças de cada um desses pacotes devem ser estudadas, e possivelmente negociadas, antes que estes pacotes possam ser incluídos em qualquer redistribuição (por exemplo, em um CD-ROM).

- `stable/contrib/`: Este diretório contém pacotes que atendem aos requerimentos da DFSG e podem ser **livremente distribuídos**, mas de alguma forma dependem de um pacote que **não** é livremente distribuível e portanto disponível somente na seção non-free.

Agora, além das localizações acima, os pacotes físicos novos são localizados sob o diretório `pool` ('O diretório `pool`' on page 9).

O estado atual dos bugs da distribuição `stable` é relatado na página web Problemas da Stable (`http://ftp-master.debian.org/testing/stable_probs.html`).

2.1.4 A distribuição `testing`

As entradas de pacotes para a distribuição `testing`, Debian Etch, são gravadas no diretório `testing` (ligação simbólica para `etch/`) depois que os mesmos tenham passado por um certo nível de testes na `unstable`. Agora, além das localizações acima, novos pacotes físicos estão localizados sob o diretório `pool` ('O diretório `pool`' on page 9). Existem também os subdiretótios `main`, `contrib`, e `non-free` na `testing`, que têm as mesmas funções como na `stable`.

Estes pacotes devem estar sincronizados em todas as arquiteturas onde eles foram compilados e não devem ter dependências que façam com que não seja possível instalá-los; eles também têm que possuir menos bugs críticos ao lançamento do que as versões atualmente na `unstable`. Dessa forma, esperamos que a `testing` esteja sempre perto de ser uma candidata ao lançamento. Maiores detalhes sobre o mecanismo da testing estão disponíveis em `http://www.debian.org/devel/testing`.

O último estado da distribuição `testing` está relatado nestes sites :

- desculpas de atualização (`http://ftp-master.debian.org/testing/update_excuses.html`)
- problemas da testing (`http://ftp-master.debian.org/testing/testing_probs.html`)
- bugs críticos ao lançamento (`http://bugs.debian.org/release-critical/`)
- bugs do sistema básico (`http://bugs.qa.debian.org/cgi-bin/base.cgi`)
- bugs em pacotes standard e em tarefas (`http://bugs.qa.debian.org/cgi-bin/standard.cgi`)
- outros bugs e notas sobre festas de matança de bugs (`http://bugs.qa.debian.org/`)

2.1.5 A distribuição `unstable`

As entradas de pacotes para a distribuição `unstable`, sempre com o codinome "Sid", são gravadas no diretório `unstable` (ligação simbólica para `sid/`) depois que é feito o upload dos mesmos para o repositório Debian, e permanecem por lá até que são movidos para `testing/`. Novos pacotes físicos estão localizados sob o diretório `pool` ('O diretório `pool`' on page 9). Existem também subdiretórios `main`, `contrib` e `non-free` em `unstable/`, que têm as mesmas funções como na `stable/`.

A distribuição `unstable` contém um snapshot do sistema de desenvolvimento mais recente. Os usuários podem usar e testar estes pacotes, mas são alertados de seu estado de preparação. A vantagem de usar a distribuição `unstable` é que você está sempre em dia no projeto de software Debian — mas se ele quebra, você tem que arrumar as coisas você mesmo :-)

O estado atual dos bugs da distribuição `unstable` é relatado na página web Problemas da Unstable (`http://ftp-master.debian.org/testing/unstable_probs.html`).

2.1.6 A distribuição `frozen`

Quando a distribuição `testing` está madura o bastante, ela é congelada, o que significa que nenhum código novo é aceito, somente correções de bugs, caso necessário. Uma nova árvore testing também é criada no diretório `dists` e lhe é atribuído um novo nome. A distribuição frozen passa por alguns meses de testes, com atualizações intermitentes e congelamentos profundos chamados "ciclos de testes".

Mantemos um registro dos bugs na distribuição frozen que podem atrasar o lançamento de um pacote ou bugs que possam atrasar o lançamento por completo. Uma vez que essa contagem de bugs seja minimizada até um valor aceitável, a distribuição frozen se torna estável, é lançada, e a distribuição stable anterior se torna obsoleta (e é movida para o repositório).

2.1.7 Codinomes das distribuições Debian

Nomes físicos de diretórios no diretório `dists/`, como `sarge/` e `etch/`, são somente "codinomes". Quando uma distribuição Debian está no estágio de desenvolvimento, ela não possui número de versão e sim um codinome. O propósito destes codinomes é tornar o espelhamento das distribuições Debian mais fácil (caso um diretório como `unstable` seja de repente mudado para `stable/`, o download de uma porção de coisas teria que ser feito de novo desnecessariamente).

Atualmente, `stable` é uma ligação simbólica para `sarge/` e `testing` é uma ligação simbólica para `etch/`. Isto significa que `Sarge` é a atual distribuição estável e `Etch` é a atual distribuição de testes.

`unstable` é uma ligação simbólica permanente para `sid/`, uma vez que Sid é sempre a distribuição instável.

2.1.8 Codinomes usados no passado

Outros codinomes que já foram usados são : "Buzz" para a versão 1.1, "Rex" para a versão 1.2, "Bo" para as versões 1.3.x, "Hamm" para a versão 2.0, "Slink" para a versão 2.1, "Potato" para a versão 2.2, "Woody" para a versão 3.0, e "Sarge" para a versão 3.1.

2.1.9 A origem dos codinomes

Até agora eles foram personagens tirados do filme *Toy Story* feito pela Pixar.

- **Buzz** (Buzz Lightyear) era o homem do espaço,
- **Rex** era o tiranossauro,
- **Bo** (Bo Peep) era a garota que cuidava do rebanho,
- **Hamm** era o porquinho cofre,
- **Slink** (Slinky Dog) era o cachorro de brinquedo,
- **Potato** era, claro, o Sr. Cabeça de Batata,
- **Woody** era o cowboy,
- **Sarge** era o líder dos Homens do Exército de Plástico Verde,
- **Etch** (Etch-a-Sketch) era o blackboard,
- **Sid** era o garoto da porta ao lado que destruía os brinquedos.

2.1.10 O diretório `pool`

Historicamente, os pacotes eram mantidos em um subdiretório de `dists` correspondendo à distribuição que os continha. Isso acabou trazendo vários problemas, como um alto consumo de banda nos espelhos quando mudanças maiores eram feitas.

Os pacotes são agoras mantidos em uma grande "piscina" (pool), estruturada de acordo com o nome do pacote fonte. Para que isso seja gerenciável, a piscina é subdividida por seções (`main`, `contrib` e `non-free`) e pela primeira letra do nome do pacote fonte. Estes diretórios contêm diversos arquivos: os pacotes binários para cada arquitetura, e os pacotes fontes a partir dos quais os pacotes binários foram gerados.

Você pode descobrir onde cada pacote é colocado executando um comando como `apt-cache showsrc` *`nomemeupacote`* e olhando na linha "Directory:". Por exemplo, os pacotes do `apache` estão armazenados em `pool/main/a/apache/`. Como há muitos pacotes `lib*`, esses são tratados especialmente: por exemplo, os pacotes `libpaper` estão armazenados em `pool/main/libp/libpaper/`.

Os diretórios `dists` ainda são usados pelos arquivos de índice usados por programas como o `apt`. Além disso, no momento que escrevo, distribuições antigas não foram convertidas para usar piscinas portanto você verá caminhos contendo nomes de distribuições como `potato` ou `woody` no campo "Diretório" do cabeçalho.

Normalmente, você não terá que se preocupar com isto, uma vez que o novo `apt` e provavelmente o antigo `dpkg-ftp` (consulte 'Métodos para atualizar um sistema Debian' on page 19) irão gerenciar isso sem problemas. Caso você queira mais informações, veja a RFC: implementation of package pools (`http://lists.debian.org/debian-devel-announce/2000/debian-devel-announce-200010/msg00007.html`).

2.1.11 Notas históricas sobre a Sid

Quando a atual Sid não existia, a organização do site do repositório Debian tinha uma grande falha: existia uma pressuposição de que quando uma arquitetura era criada no `unstable/` atual, ela seria lançada quando esta distribuição virasse a nova `stable`. Para muitas arquiteturas esse não era o caso, e como resultado todos aqueles diretórios tiveram de ser movidos na época de lançamento. Isto era impraticável pois a movimentação comeria muita banda.

Os administradores do repositório resolveram este problema por diversos anos colocando binários para arquiteturas não lançadas em um diretório especial chamado `sid`. Para aquelas arquiteturas ainda não lançadas, a primeira vez que as mesmas foram lançadas existia uma ligação do atual `stable/` para `sid/`, e a partir de então elas eram criadas dentro da árvore `unstable/` como de costume. Este layout era de certa forma confuso para os usuários.

Com o advento das piscinas de pacotes (consulte 'O diretório `pool`' on the preceding page) durante o desenvolvimento da distribuição Woody, pacotes binários começaram a ser armazenados em uma localização canônica na piscina, não importando a distribuição, assim lançar uma distribuição não mais causava grade consumo de banda nos espelhos (existe, porém, uma porção de consumo gradual de banda durante o processo de desenvolvimento).

2.1.12 Pacotes enviados para `incoming/`

Os pacotes enviados estão primeiro localizados em `http://incoming.debian.org/` antes de serem checados para assegurar que eles realmente vieram de um desenvolvedor Debian (e são colocados no subdiretório `DELAYED` caso tenham sido enviados por um não-desenvolvedor (Non-Mantainer Upload - NMU)). Uma vez por dia, eles são movidos de `incoming/` para `unstable/`.

Em uma emergência, você pode querer instalar pacotes de `incoming/` antes que eles atinjam `unstable/`.

2.1.13 Obtendo um pacote antigo

Enquanto a maioria das distribuições Debian atuais são mantidas sob o diretório `debian` em cada site espelho Debian (`http://www.debian.org/mirror/`), repositórios para distribuições Debian mais antigas como a Slink são mantidos em `http://archive.debian.org/` ou sob o diretório `debian-archive` em cada site espelho Debian.

Os pacotes antigos da `testing` e da `unstable` podem ser encontrados em `http://snapshot.debian.net/`.

2.1.14 Seções de Arquiteturas

Dentro de cada árvore maior de diretórios (`dists/stable/main`, `dists/stable/contrib`, `dists/stable/non-free`, `dists/unstable/main/`, etc.), as entradas do pacote binário residem em subdiretórios cujos nomes indicam a arquitetura do chip para o qual eles foram compilados.

- `binary-all/`, para pacotes que são independentes de arquitetura. Estes incluem, por exemplo, scripts Perl ou documentação pura.

- `binary-`*`plataforma`*`/`, para pacotes que executam em uma plataforma binária particular.

Por favor note que os pacotes binários atuais para a `testing` e `unstable` não mais residem nestes diretórios, mas em um diretório de alto nível `pool`. Os arquivos de índice (`Packages` e `Packages.gz`) foram mantidos, porém, para compatibilidade anterior.

Para conhecer as arquiteturas binárias suportadas atualmente, consulte as Notas de Lançamento para cada distribuição. Elas podem ser localizadas nos sites das Notas de Lançamento para a stable (`http://www.debian.org/releases/stable/releasenotes`) e para a testing (`http://www.debian.org/releases/testing/releasenotes`).

2.1.15 O código-fonte

O código-fonte está incluído para tudo no sistema Debian. Além disso, os termos das licenças da maioria dos programas no sistema **requerem** que o código-fonte seja distribuído junto com os programas, ou que uma oferta para fornecer o código-fonte acompanhe os programas.

Normalmente o código-fonte é distribuído nos diretórios `source`, os quais são paralelos a todos os diretórios de binários para arquiteturas específicas ou, mais recentemente, no diretório `pool` (consulte 'O diretório `pool`' on page 9). Para obter o código-fonte sem ter estar familiarizado com a estrutura do repositório Debian , tente um comando como `apt-get source` *`meunomedepacote`*.

Alguns pacotes, notavelmente `pine`, estão somente disponíveis como pacotes fonte devido a suas limitações de licenciamento. (Recentemente o pacote `pine-tracker` foi fornecido para facilitar a instalação do Pine.) Os procedimentos descritos em 'Portar um pacote para o sistema `stable`' on page 95 e 'Empacotamento' on page 243 fornecem meios para construir um pacote manualmente.

O código-fonte pode ou não estar disponível para pacotes nos diretórios `contrib` e `non-free`, os quais não são formalmente parte do sistema Debian.

2.2 O sistema de gerenciamento de pacotes Debian

2.2.1 Visão Geral dos pacotes Debian

Pacotes geralmente contêm todos os arquivos necessários para implementar um conjunto de comandos relacionados ou recursos. Existem dois tipos de pacotes Debian :

- **Pacotes binários**, os quais contém executáveis, arquivos de configuração, páginas de manual/info, informação de copyright e outra documentação. Estes pacotes são distribuídos em um formato de arquivo específico Debian (consulte 'Formato de pacotes Debian' on the following page); eles são normalmente identificados por terem uma extensão de arquivo `.deb`. Pacotes binários podem ser desempacotados usando o utilitário Debian `dpkg`; detalhes são dados em sua página de manual.
- **Pacotes fonte**, os quais consistem de um arquivo `.dsc` descrevendo o pacote fonte (incluindo os nomes dos arquivos a seguir), um arquivo `.orig.tar.gz` que contém o

fonte original não modificado em formato tar e compactado com o gzip, e normalmente um arquivo `.diff.gz` que contém as mudanças específicas do Debian feitas em relação ao fonte original. O utilitário `dpkg-source` empacota e desempacota arquivos fonte Debian; detalhes são fornecidos em sua página de manual.

A instalação de software pelo sistema de pacotes utiliza "dependências" que são cuidadosamente especificadas pelos mantenedores dos pacotes. Estas dependências estão documentadas no arquivo `control` associado a cada pacote. Por exemplo, o pacote contendo o compilador GNU C (`gcc`) "depende" do pacote `binutils` que inclui o ligador e o montador. Caso um usuário tente instalar o `gcc` sem ter instalado primeiro o `binutils`, o sistema de gerenciamento de pacotes (dpkg) emitirá uma mensagem de erro dizendo que ele precisa também do `binutils` e parará de instalar o `gcc`. (Porém, esta facilidade pode ser circulada pelo usuário insistente, consulte `dpkg(8)`.) Para detalhes adicionais, consulte 'Dependências de pacotes' on page 15 abaixo.

As ferramentas de empacotamento Debian podem ser usadas para :

- manipular e gerenciar pacotes ou partes de pacotes,
- auxiliar o usuário na separação de pacotes que devem ser transmitidos através de uma mídia de tamanho limitado, como disquetes,
- auxiliar desenvolvedores na construção de repositórios de pacotes, e
- auxiliar usuários na instalação de pacotes que se encontram em um site repositório Debian.

2.2.2 Formato de pacotes Debian

Um "pacote" Debian, ou um arquivo Debian, contém os arquivos executáveis, bibliotecas e documentação associada a um programa em particular ou a uma suíte ou conjunto de programas relacionados. Normalmente, um arquivo Debian tem um nome de arquivo finalizando em `.deb`. [1]

Os detalhes do formato de pacote binário Debian estão descritos na página de manual `deb(5)`. Devido a este formato interno estar sujeito a mudanças (entre versões maiores do Debian), sempre utilize o `dpkg-deb(1)` para manipular arquivos `.deb`.

Pelo menos para a distribuição Sarge, todos os arquivos Debian são manipuláveis pelos comandos Unix padrões `ar` e `tar`, mesmo quando comandos dpkg não estão disponíveis.

---

[1] O projeto `debian-installer` introduziu nomes de arquivos de pacotes que terminam em `.udeb`. Resumidamente, isso é um formato micro-`.deb` que não precisa seguir a política Debian exatamente, faltando conteúdos como documentação e é dedicado a ser usado apenas pelo `debian-installer`, o novo instalador Debian que está sendo desenvolvido para o lançamento do Sarge. Os formatos de arquivo `.deb` e `.udeb` são idênticos. O programa `udpkg` usado para manipular pacotes `.udeb` tem capacidade mais limitada que o `dpkg` padrão e suporta menos relacionamentos entre pacotes. A diferença no nome é porque os mantenedores de repositórios Debian não estavam contentes com `.debs` no repositório que não seguiam a política, então um nome diferente foi escolhido para destacar isso e diminuir a possibilidade das pessoas instalá-los inconscientemente em sistemas reais. Os `.udebs` são usados em um ramdisk inicial durante a instalação básica apenas para criar um sistema Debian muito restrito.

2.2.3 Convenções de nomenclatura para nomes de arquivos de pacotes Debian

Os nomes de arquivos de pacotes Debian seguem a seguinte convenção :

```
foo_NúmeroDeVersão-NúmeroDeRevisãoDebian.deb
```

onde *foo* representa o nome do pacote. Como uma checagem, pode-se determinar o nome do pacote associado a um arquivo Debian em particular (arquivo `.deb`) em uma das seguintes maneiras :

- inspecione o arquivo "Packages" no diretório onde o mesmo foi armazenado em um site repositório Debian . Este arquivo contém uma estrofe descrevendo cada pacote; o primeiro campo de cada estrofe é o nome formal do pacote.
- use o comando `dpkg --info foo_VVV-RRR.deb` (onde *VVV* e *RRR* são a versão e a revisão do pacote em questão, respectivamente). Isto exibe, entre outras coisas, o nome do pacote do arquivo sendo desempacotado.

O componente *VVV* é o número de versão especificado pelo desenvolvedor original (upstream). Não existem padrões ditando números de versão, portanto eles podem ter formatos tão diferentes como "19990513" e "1.3.8pre1".

O componente *RRR* é um número de revisão Debian e é especificado pelo desenvolvedor Debian (ou um usuário individual caso o mesmo escolha construir o pacote ele mesmo). Este número corresponde ao nível de revisão do pacote Debian; portanto, um novo nível de revisão geralmente significa mudanças no Makefile Debian (`debian/rules`), no arquivo de controle Debian (`debian/control`), nos scripts de instalação ou remoção (`debian/p*`) ou nos arquivos de configuração usados pelo pacote.

2.2.4 Preservação da configuração local

A preservação dos arquivos configuráveis pelo usuário é habilitada através do mecanismo "conffiles" do Debian. Arquivos de configuração do usuário (normalmente colocados em `/etc/`) são especificados no `conffiles` dentro do sistema de pacotes Debian. O sistema de gerenciamento de pacotes garante a não sobreescrita destes arquivos quando o pacote é atualizado.

Para determinar exatamente quais arquivos são preservados durante uma atualização, execute :

```
dpkg --status pacote
```

e olhe em "Conffiles:".

Detalhes sobre o conteúdo de um arquivo `conffiles` Debian são fornecidos no Manual de Políticas Debian, seção 11.7 (consulte 'Referências' on page 249).

2.2.5 Scripts de manutenção Debian

Os scripts de manutenção Debian são scripts executáveis que são automaticamente executados antes ou depois que um pacote é instalado. Junto com um arquivo de nome `control`, todos esses arquivos são parte da seção “control” de um arquivo Debian.

Os arquivo individuais são:

preinst Este script é executado antes que seu pacote seja desempacotado de seu arquivo Debian (`.deb`). Muitos scripts “preinst” param serviços para os pacotes que estão sendo atualizados até que sua instalação ou atualização esteja completa (seguindo a execução com sucesso do script “postinst”).

postinst Este script tipicamente completa qualquer configuração requerida de um pacote uma vez que o mesmo tenha sido desempacotado de seu arquivo Debian (`.deb`). Geralmente, scripts 'postinst' fazem perguntas aos usuários e/ou avisam o usuário que caso o mesmo aceite valores padrões ele deverá se lembrar de voltar e reconfigurar o pacote conforme a necessidade. Muitos scripts “postinst” executam então quaisquer comandos necessários para iniciar ou reiniciar um serviço uma vez que o novo pacote foi instalado ou atualizado.

prerm Este script tipicamente pára quaisquer daemons que estão associados com um pacote. Ele é executado antes da remoção de arquivos associados com um pacote.

postrm Este script tipicamente modifica ligações ou outros arquivos associados com um pacote e/ou remove arquivos criados pelo pacote. (Consulte também ‘Pacotes virtuais’ on the next page.)

Atualmente todos os arquivos de controle podem ser encontrados no `/var/lib/dpkg/info`. Os arquivos relevantes ao pacote `foo` iniciam com o nome “foo” e possuem extensões de arquivos “preinst”, “postinst”, etc, de acordo. O arquivo `foo.list` neste diretório lista todos os arquivos que foram instalados com o pacote `foo`. (Note que a localização destes arquivos é algo interno do dpkg, e pode estar sujeita a mudanças.)

2.2.6 Prioridade de pacotes

A cada pacote Debian é atribuído uma **prioridade** pelos mantenedores da distribuição, como um auxílio ao sistema de gerenciamento de pacotes. As prioridades são :

- Pacotes **Required** são necessários para o funcionamento correto do sistema.

  Isso inclui todas as ferramentas necessárias para reparar defeitos no sistema. Você não deve remover esses pacotes ou seu sistema pode vir a se tornar completamente quebrado e você poderá provavelmente não ser capaz de usar o dpkg para colocar as coisas de volta em seus lugares. Sistemas somente com pacotes Required são provavelmente inutilizáveis para muitos propósitos, mas possuem funcionalidade o suficiente para permitir que o administrador do sistema inicie o sistema e instale mais software.

- Pacotes **Important** devem ser encontrados em qualquer sistema parecido com Unix.

  Outros pacotes sem os quais o sistema não funcionará bem ou não será muito útil se encaixam aqui. Isso **não** inclui o Emacs ou X11 ou TeX ou quaisquer outras grandes aplicações. Esses pacotes constituem somente a infraestrutura mínima.

- Pacotes **Standard** são padrão em qualquer sistema Linux, incluindo um sistema em modo caracter razoavelmente pequeno mas não muito limitado.

  Isso é o que será instalado por padrão caso os usuários não selecionem mais nada. Isso não inclui muitas grandes aplicações, mas inclui o Emacs (ele é mais uma peça da infraestrutura do que uma aplicação) e um subconjunto razoável do Tex e LaTeX (caso isso seja possível sem o X).

- Pacotes **Optional** incluem todos aqueles pacotes que você poderia razoavelmente querer instalar mesmo caso não esteja familiarizado com eles e caso você não possua requerimentos especializados.

  Isso inclui X11, uma distribuição completa TeX e uma porção de aplicações.

- Pacotes **Extra** são pacotes que conflitam com outros pacotes com prioridades mais altas provavelmente, terão pouco uso para usuários que não são familiarizados a eles, ou têm requerimentos especiais que não os deixam ser encaixados em "Optional".

2.2.7 Pacotes virtuais

Um pacote virtual é um nome genérico que se aplica a qualquer um de um grupo de pacotes, todos os quais oferecem funcionalidade básica similar. Por exemplo, os programas `tin` e `trn` são leitores de notícias e devem portanto satisfazer quaisquer dependências de um programa que requer um leitor de notícias no sistema para que funcione ou seja útil. É portanto dito que eles fornecem um "pacote virtual" chamado `news-reader`.

Similarmente, o `exim` e o `sendmail` oferecem ambos a funcionalidade de um agente de transporte de mensagens, É portanto dito que eles fornecem o pacote virtual `mail-transport-agent`. Caso um dos dois esteja instalado, então qualquer programa que depende da instalação de um agente de transporte de mensagens estará satisfeito pela existência desse pacote virtual.

O Debian possui um mecanismo que, se mais de um pacote que fornece o mesmo pacote virtual estiver instalado em um sistema, o administrador do sistema pode definir um como o pacote preferido. O comando relevante é `update-alternatives` e é descrito mais adiante em 'Comandos alternativos' on page 98.

2.2.8 Dependências de pacotes

O sistema de pacotes Debian possui uma faixa de "dependências" de pacotes que é desenvolvida para indicar (com uma sinalização única) o nível no qual o Programa A pode operar independentemente da existência do Programa B em um dado sistema :

- O Pacote A **depende** do Pacote B se B precisar absolutamente estar instalado para que A possa ser executado. Em alguns casos, A depende não somente de B, mas de uma versão específica de B. Nestes casos, a dependência da versão é normalmente um limite mínimo, no sentido de que A depende em qualquer versão de B mais recente do que alguma versão especificada.
- O Pacote A **recomenda** o Pacote B caso o mantenedor do pacote julgue que a maioria dos usuários não iriam querer A sem também ter a funcionalidade fornecida por B.
- O Pacote A **sugere** o Pacote B caso B contenha arquivos que estejam relacionados a (e normalmente melhorem) a funcionalidade de A.
- O Pacote A **conflita** com o Pacote B quando A não funciona caso B esteja instalado no sistema. Na maioria das vezes, conflitos são casos onde A contém arquivos que são uma melhora daqueles em B. "Conflicts" (conflita) são normalmente combinados com "replaces" (substitui).
- O Pacote A **substitui** o Pacote B quando arquivos instalados por B são removidos e (em alguns casos) sobrescritos por arquivos em A.
- O Pacote A **fornece** o Pacote B quando todos os arquivos e as funcionalidades de B estão incorporadas em A. Este mecanismo oferece uma maneira para usuários com pouco espaço em disco obterem somente aquela parte do pacote A que realmente precisam.

Informação mais detalhada sobre o uso de cada um desses termos pode ser encontrada no *Manual de Empacotamento* e no *Manual de Políticas*.

Note que o `dselect` possui um controle mais refinado sobre os pacotes especificados por **recomendações** e **sugestões** do que o `apt-get`, o qual simplesmente pega todos os pacotes especificados como **dependências** e deixa para trás os pacotes especificados como **recomendados** e **sugestões**. Ambos os programas em sua forma moderna usam o APT como motor.

2.2.9 O significado de Pré-dependências

"Pre-depends" (pré-dependências) é uma dependência especial. No caso de um pacote comum, o `dpkg` irá desempacotar seu arquivo (ou seja, seu arquivo `.deb`) independentemente dos arquivos dos quais este depende existirem ou não no sistema. Desempacotar significa que o `dpkg` irá extrair os arquivos de seu arquivo que foi criado para ser instalado em seu sistema e colocá-los nos lugares. Caso estes pacotes **dependam** da existência de outros pacotes em seu sistema, o `dpkg` se recusará a completar a instalação (executando sua ação "configure") até que os outros pacotes sejam instalados.

Porém, para alguns pacotes, o `dpkg` se recusará até mesmo a desempacotá-los até que certas dependências estejam resolvidas. É dito que tais pacotes "pré-dependem" da presença de outro(s) pacote(s). O projeto Debian forneceu este mecanismo para suportar a atualização segura do formato `a.out` para o formato `ELF`, onde a ordem na qual os pacotes eram desempacotados era crítica. Existem outras situações de grandes atualizações onde este método é útil, por exemplo, para pacotes com prioridade "required" e sua dependência libc.

Novamente, informação mais detalhada sobre isso pode ser encontrada no *Manual de Empacotamento*.

2.2.10 Estado do pacote

O estado do pacote pode ser "unknow", "install", "remove", "purge" ou "hold". Estas flags "want" dizem o que o usuário quis fazer com o pacote (como indicado pelas ações do usuário na seção "Selecionar" do `dselect` ou pelas invocações diretas do `dpkg`).

Seus significados são :

- **unknown** - o usuário nunca indicou se ele queria o pacote.
- **install** - o usuário quer o pacote instalado ou atualizado.
- **remove** - o usuário quer o pacote removido, mas não quer remover quaisquer arquivos de configuração.
- **purge** - o usuário quer que o pacote seja removido completamente, incluindo os arquivos de configuração.
- **hold** - o usuário quer que esse pacote não seja processado, ou seja, ele quer manter a versão atual no estado atual seja este qual for.

2.2.11 Evitando que pacotes sejam atualizados

Existem dois mecanismos para evitar que pacotes sejam atualizados : através do `dpkg` ou, começando no Woody, através do APT.

Com o `dpkg`, exporte primeiro a lista de seleções de pacotes :

```
dpkg --get-selections \* > selections.txt
```

Edite então o arquivo resultante *`selections.txt`* mudando a linha contendo o pacote que você desejaria evitar que fosse atualizado, por exemplo, `libc6`, disso:

```
libc6                         install
```

para isso :

```
libc6                         hold
```

Salve o arquivo e recarregue-o na base de dados do `dpkg` usando :

```
dpkg --set-selections < selections.txt
```

Ou, caso você conheça o nome do pacote a ser mantido (hold), simplesmente execute :

```
echo libc6 hold | dpkg --set-selections
```

Este processo mantém pacotes no processo de instalação de cada arquivo de pacote.

O mesmo efeito pode ser obtido através do `dselect`. Simplesmente entre na tela [S]elecionar, encontre o pacote que você deseja manter em seu estado atual e pressione a tecla '=' (ou 'H'). As modificações serão válidas imediatamente depois que você sair da tela [S]elecionar.

O sistema APT na distribuição Woody possui um novo mecanismo alternativo para manter pacotes durante o processo de obtenção do arquivo usando `Pin-Prioriry`. Consulte a página de manual `apt_preferences(5)`,e também `http://www.debian.org/doc/manuals/apt-howto/` ou o pacote `apt-howto`; 'Visão geral do arquivo `/etc/apt/preferences`' on page 87 também contém uma explicação breve.

2.2.12 Pacotes fonte

Pacotes fonte são distribuídos em um diretório chamado `source` e você pode fazer o download dos mesmos manualmente ou use

```
apt-get source foo
```

para obtê-los (consulte a página de manual `apt-get(8)` para maiores informações sobre como configurar o APT para fazer isso).

2.2.13 Construindo pacotes binários a partir de um pacote fonte

Para um pacote *foo*, você precisará de todos os arquivos *foo_**`.dsc`, *foo_**`.tar.gz` e *foo_*.diff*`.gz` para compilar o fonte (nota: não existe `.diff.gz` para um pacote nativo Debian).

Uma vez que você os tenha, caso você possua o pacote `dpkg-dev` instalado, o comando

```
dpkg-source -x foo_versão-revisão.dsc
```

irá extrair o pacote em um diretório chamado *foo-versão*.

Execute os seguintes comandos para criar o pacote binário:

```
$ cd foo-versão
$ su -c "apt-get update ; apt-get install fakeroot"
$ dpkg-buildpackage -rfakeroot -us -uc
```

E então,

```
# su -c "dpkg -i ../foo_versão-revisão_arquit.deb"
```

para instalar o novo pacote construído. Veja 'Portar um pacote para o sistema `stable`' on page 95.

2.2.14 Criando novos pacotes Debian

Para uma descrição mais detalhada, leia o *Guia dos Novos Mantenedores*, disponível no pacote `maint-guide` ou em `http://www.debian.org/doc/manuals/maint-guide/`.

2.3 Atualizando um sistema Debian

Um dos objetivos do Debian é oferecer um caminho de atualização consistente e um processo de atualização seguro, e nós sempre fazemos o máximo possível para que uma nova versão seja facilmente atualizável para quem atualiza da versão anterior. Pacotes irão alertar o usuário quando existirem notícias importantes durante o processo de atualização e irão frequentemente oferecer uma solução para um possível problema.

Você deve também ler as Notas de Lançamento, o documento que descreve os detalhes de atualizações específicas, fornecido em todos os CDs Debian e disponível na WWW em `http://www.debian.org/releases/stable/releasenotes` ou `http://www.debian.org/releases/testing/releasenotes`.

Um guia prático para atualizações está disponível em 'Gerenciamento de pacotes Debian' on page 81. Esta seção descreve os detalhes fundamentais.

2.3.1 Métodos para atualizar um sistema Debian

É possível simplesmente executar uma chamada FTP anônima ou usar o `wget` em um repositório Debian, procurar atentamente até que seja encontrado o arquivo desejado, fazer o download do mesmo e finalmente instalá-lo usando o `dpkg`. Note que o `dpkg` irá instalar arquivos de atualização em seus lugares, mesmo em um sistema em execução. Algumas vezes, um pacote revisado irá requerer a instalação de uma nova versão revisada de outros pacotes e neste caso a instalação irá falhar até que/a menos que os outros pacotes sejam instalados.

Muitas pessoas acham que este método consome muito tempo, uma vez que o Debian se desenvolve tão rapidamente — tipicamente, uma dúzia ou mais pacotes são disponibilizados toda semana. Este número é maior logo antes de um lançamento de uma versão maior. Para lidar com esta avalanche, muitas pessoas preferem usar um programa automatizado para atualizar. Diversas ferramentas de gerenciamento de pacotes especializadas estão disponíveis para este propósito.

2.3.2 Visão geral das ferramentas de gerenciamento de pacotes

O sistema de gerenciamento de pacotes possui dois objetivos : a manipulação do arquivo de pacote propriamente dito e obtenção de arquivos de pacotes do repositório Debian. O `dpkg` executa a primeira tarefa e o APT e o `dselect` a última.

2.3.3 `dpkg`

Este é o principal programa para manipular arquivos de pacotes; leia `dpkg(8)` para um descrição completa.

O `dpkg` é fornecido com diversos programas suplementares primitivos.

- dpkg-deb: Manipula arquivos `.deb`. `dpkg-deb(1)`
- dpkg-ftp: Um antigo comando de obtenção de arquivos de pacotes. `dpkg-ftp(1)`
- dpkg-mountable: Um antigo comando de obtenção de arquivos de pacotes. `dpkg-mountable(1)`
- dpkg-split: Divide um pacote grande em arquivos menores. `dpkg-split(1)`

O `dpkg-ftp` e `dpkg-mountable` ficaram obsoletos com a introdução do sistema APT.

2.3.4 APT

O APT (Advanced Packaging Tool) é uma avançada interface para o sistema de gerenciamento de pacotes Debian, consistindo de vários programas cujos nomes tipicamente começam com "apt-". O `apt-get`, `apt-cache` e o `apt-cdrom` são ferramentas de linha de comando para gerenciar pacotes. Eles também funcionam como programas "back-end" do usuário para outras ferramentas, como o `dselect` e o `aptitude`.

Para maiores informações, instale o pacote `apt` e leia `apt-get(8)`, `apt-cache(8)`, `apt-cdrom(8)`, `apt.conf(5)`, `sources.list(5)`, `apt_preferences(5)` (Woody), e `/usr/share/doc/apt/guide.html/index.html`.

Uma fonte alternativa de informação é o APT HOWTO (`http://www.debian.org/doc/manuals/apt-howto/`). Este pode ser instalado pelo pacote `apt-howto` em `/usr/share/doc/Debian/apt-howto/`.

`apt-get upgrade` e `apt-get dist-upgrade` instalam apenas os pacotes listados em "Depends:" e ignoram os pacotes listados em "Recommends:" e "Suggests:". Para evitar isso use o `dselect`.

2.3.5 `dselect`

Este programa é uma interface de usuário baseada em menus para o sistema de gerenciamento de pacotes Debian. Ele é particularmente útil para primeiras instalações e atualizações em larga escala. Veja '`dselect`' on page 83

Para maiores informações, instale o pacote `install-doc` e leia `/usr/share/doc/install-doc/dselect-beginner.en.html` ou Documentação para Iniciantes do dselect (`http://www.debian.org/releases/woody/i386/dselect-beginner`).

2.3.6 Atualizando um sistema em execução

O kernel (sistema de arquivos) em sistemas Debian suporta a troca de arquivos mesmo quando estes estão sendo usados.

Oferecemos também um programa chamado `start-stop-daemon` que é usado para iniciar daemons em tempo de inicialização da máquina ou para parar daemons quando o nível de execução do kernel é mudado (exemplo, de multiusuário para usuário único ou para halt). O mesmo programa é usado pelos scripts de instalação quando um novo pacote contendo um daemon é instalado, para parar daemons em execução, e reiniciá-los quando necessário.

Note que o sistema Debian não requer o uso do modo de usuário único para atualizar um sistema em execução.

2.3.7 Arquivos `.deb` baixados e em cache

Caso você tenha feito o download de arquivos para seu disco (o que não é absolutamente necessário, veja acima a descrição de `dpkg-ftp` ou APT), depois de instalar os pacotes você pode removê-los de seu sistema.

Caso o APT seja usado, esses arquivos são colocados em cache no diretório `/var/cache/apt/archives/`. Você pode apagá-los depos da instalação (`apt-get clean`) ou copiá-los para o diretório `/var/cache/apt/archives/` de outra máquina para economizar tempo de download durante instalações subseqüentes.

2.3.8 Mantendo registros para atualizações

O `dpkg` mantém um registro dos pacotes que foram desempacotados, configurados, removidos e/ou expurgados, mas não mantém (atualmente) um log da atividade do terminal que ocorreu enquanto um pacote esteve sendo manipulado.

A maneira mais simples de contornar isso é executar suas sessões `dpkg`, `dselect`, `apt-get`, etc., dentro do programa `script(1)`.

2.4 O processo de inicialização Debian

2.4.1 O programa `init`

Como todos os Unices, o Debian inicia executando o programa `init`. O arquivo de configuração para o `init` (que é `/etc/inittab`) especifica que o primeiro script a ser executado deve ser o `/etc/init.d/rcS`. Esse script executa todos os scripts em `/etc/rcS.d/` através do source ou fork de subprocessos, dependendo de sua extensão de arquivo, para executar a inicialização como a checagem e a montagem de sistemas de arquivos, carregamento de módulos, início de serviços de rede, configuração do relógio, etc. Então, por compatibilidade, ele também executa os arquivos (exceto aqueles com um '.' em seu nome) em `/etc/rc.boot`. Quaisquer scripts no diretório posterior são normalmente reservados para o uso do administrador do sistema e usá-los em pacotes é obsoleto. Veja 'Inicialização do sistema' on page 137 e Níveis de execução de Sistema e scripts init.d (`http://www.debian.org/doc/debian-policy/ch-opersys#s-sysvinit`) no Manual de Políticas Debian para maiores informações.

2.4.2 Níveis de execução

Depois de completar o processo de inicialização, o `init` executa todos os scripts de inicialização em um diretório especificado pelo nível de execução padrão (este nível de execução é dado pela entrada para o `id` em `/etc/inittab`. Como a maioria dos Unices compatíveis com System V, o Linux possui 7 níveis de execução :

- 0 (parar o sistema),
- 1 (modo de usuário único),
- 2 through 5 (vários modos multiusuário), e
- 6 (reinicia o sistema).

Sistemas Debian vêm com o valor `id=2`, o que indica que o nível de execução padrão será 2 quando o estado multiusuário for iniciado e que os scripts em `/etc/rc2.d/` serão executados.

De fato, os scripts em quaisquer dos diretórios em `/etc/rcN.d/` são apenas ligações simbólicas que apontam para scripts em `/etc/init.d/`. Porém, os **nomes** dos arquivos em cada um dos diretórios `/etc/rcN.d/` são selecionados para indicar a **maneira** que os scripts em `/etc/init.d/` serão executados. Especificamente, antes de entrar em qualquer nível de execução, todos os scripts iniciados com 'K' são executados; esses scripts matam (param) serviços. Então todos os scripts iniciados com 'S' são executados; esses scripts iniciam serviços. O número de dois dígitos seguido de 'K' ou 'S' indica a ordem na qual o script é executado. Scripts de menor valor numérico são executados primeiro.

Esse método funciona porque todos os scripts em `/etc/init.d/` aceitam um argumento que pode ser "start" (iniciar), "stop" (parar), "reload" (recarregar), "restart" (reiniciar) ou "force-reload" (forçar-recarregar) e irão portanto cumprir a tarefa indicada pelo argumento. Esses scripts podem ser usados mesmo depois que um sistema tenha sido iniciado para controlar vários processos.

Por exemplo, com o argumento "reload" o comando

```
# /etc/init.d/exim4 reload
```

envia ao daemon exim4 um sinal para que o mesmo releia seu arquivo de configuração.

2.4.3 Personalizando o processo de inicialização

O Debian não utiliza o diretório `rc.local` no estilo BSD para personalizar o processo de inicialização; ao invés disso ele fornece o seguinte mecanismo de personalização.

Suponha que um sistema precisa executar o script `foo` na inicialização da máquina ou ao entrar em um nível de execução (System V) em especifíco. O administrador do sistema deverá então :

1 Colocar o script `foo` dentro do diretório `/etc/init.d/`.

2 Executar o comando Debian `update-rc.d` com os argumentos apropriados para criar as ligações entre os diretórios (especificados na linha de comando) `rc?.d` e `/etc/init.d/foo`. Aqui, *?* é um número de 0 a 6 que corresponde a um dos níveis de execução System V.

3 Reiniciar o sistema.

O comando `update-rc.d` criará as ligações entre os arquivos nos diretórios `rc?.d` e o script em `/etc/init.d/`. Cada ligação iniciará com um 'S' ou um 'K', seguido por um número, seguido pelo nome do script. Quando o sistema entra em um nível de execução *N*, scripts que iniciam com 'K' em `/etc/rcN.d/` são executados com `stop` como seu argumento, seguido por aqueles começando com 'S' em `/etc/rcN.d` com `start` como seu argumento.

Alguém poderia, por exemplo, fazer com que o script `foo` seja executado na inicialização do sistema colocando-o em `/etc/init.d/` e instalando as ligações com o comando `update-rc.d foo defaults 19`. O argumento `defaults` se refere aos níveis de execução padrões, que são do nível 2 até o nível 5. O argumento `19` assegura que `foo` seja chamado antes de quaisquer scripts contendo números 20 ou superiores.

2.5 Suportando diversidades

O Debian suporta diversas maneiras de acomodar os desejos do administrador do sistema sem prejudicar o sistema.

- `dpkg-divert`, veja 'O comando `dpkg-divert`' on page 98.
- `equivs`, veja 'O pacote `equivs`' on page 98.
- `update-alternative`, veja 'Comandos alternativos' on page 98.
- `make-kpkg` pode acomodar muitos gerenciadores de inicialização. Consulte `make-kpkg(1)` e 'Método Debian padrão' on page 101.

Quaisquer arquivos sob `/usr/local/` pertencem ao administrador do sistema e o Debian não irá tocá-los. A maioria (ou todos) dos arquivos sob `/etc` são `conffiles` e o Debian não irá sobrescrevê-los em atualizações a menos que o administrador do sistema explicitamente peça isso.

2.6 Internacionalização

O sistema Debian é internacionalizado e oferece suporte para a exibição e entrada de caracteres em muitos idiomas, seja no console ou sob o X. Muitos documentos, páginas de manual e mensagens do sistema foram traduzidos para um número crescente de idiomas. Durante a instalação o Debian pede ao usuário para escolher um idioma para ser usado na instalação (e algumas vezes o variante local do idioma).

Caso seu sistema instalado não suporte todos os recursos do idioma que você precisa ou caso você precise mudar entre idiomas ou instalar um teclado diferente para suportar seu idioma, consulte 'Localização' on page 171.

2.7 Debian e o kernel

Consulte 'O kernel Linux no Debian' on page 101.

2.7.1 Compilando um kernel a partir de um fonte não Debian

Você precisa entender o política Debian em relação a cabeçalhos.

As bibliotecas C Debian são construídas com as versões **estáveis** mais atuais dos cabeçalhos do **kernel**.

Por exemplo, a versão Debian-1.2 usou a versão 5.4.13 dos cabeçalhos. Esta prática contrasta com os pacotes fontes do kernel Linux distribuídos em todos os repositórios de sites FTP Linux, que usam as versões mais recentes até mesmo dos cabeçalhos. Os cabeçalhos do kernel distribuídos com os fontes do kernel estão localizados em `/usr/include/linux/include/`.

Caso você precise compilar um programa com cabeçalhos de kernel que sejam mais novos do que aqueles fornecidos pelo pacote `lib6-dev`, você deve então adicionar `-I/usr/src/linux/include/` a sua linha de comando quando compilar. Isto ocorreu em um momento, por exemplo, com o empacotamento do daemon automounter (`amd`). Quando novos kernels mudaram internamente em relação a lidar com NFS, o `amd` precisava saber disso. Para isso foi necessário incluir os últimos cabeçalhos do kernel.

2.7.2 Ferramentas para construir kernels personalizados

Usuários que desejam (ou precisam) construir um kernel personalizado são encorajados a fazer o download do pacote `kernel-package`. Este pacote contém o script para construir o pacote do kernel e oferece a capacidade de criar um pacote kernel-image Debian somente executando o comando

```
# make-kpkg kernel_image
```

no diretório de nível principal dos fontes do kernel. Ajuda é fornecida executando o comando

```
# make-kpkg --help
```

e através da página de manual `make-kpkg(1)` e 'O kernel Linux no Debian' on page 101.

Os usuários devem fazer o download separadamente do código fonte do kernel mais atual (ou o kernel de sua escolha) a partir de seu site repositório Linux favorito, a menos que um pacote kernel-source-*versão* esteja disponível (onde *versão* significa a versão do kernel). O script de inicialização `initrd` Debian requer um patch de kernel especial chamado `initrd`; consulte `http://bugs.debian.org/149236`.

Informações detalhadas para o uso do pacote `kernel-package` são fornecidas no arquivo `/usr/doc/kernel-package/README`.

2.7.3 Condições especiais para lidar com módulos

O pacote Debian `modconf` oferece um script shell (`/usr/sbin/modconf`) que pode ser usado para personalizar a configuração dos módulos. Este script apresenta uma interface baseada em menus, perguntando ao usuário por detalhes sobre os controladores de dispositivos carregáveis em seu sistema. As respostas são usadas para personalizar o arquivo `/etc/modules.conf` (que lista apelidos e outros argumentos que devem ser usados em conjunto com outros módulos) através de arquivos em `/etc/modutils/` e `/etc/modules` (que listam os módulos que devem ser carregados em tempo de inicialização da máquina).

Como os (novos) arquivos `Configure.help` que estão agora disponíveis para suportar a construção de kernels personalizados, o pacote `modconf` vem com uma série de arquivos de ajuda (em `/usr/share/modconf/`) que oferecem informações detalhadas para cada um dos módulos. Consulte 'O kernel 2.4 modularizado' on page 103 para ver exemplos.

2.7.4 Desinstalação de um kernel antigo

O script `kernel-image-NNN.prerm` checa se o kernel que está em execução atualmente é o mesmo que o kernel que você está tentando desinstalar. Portanto você pode remover pacotes de imagens kernel indesejáveis com segurança usando esse comando :

```
# dpkg --purge --force-remove-essential kernel-image-NNN
```

(É claro, substitua *NNN* pela versão de seu kernel e seu número de revisão)

Capítulo 3

Dicas para instalação do Sistema Debian

A documentação oficial para instalação da Debian se encontra em `http://www.debian.org/releases/stable/`, e `http://www.debian.org/releases/stable/installmanual`.

As versões em desenvolvimento são encontradas em `http://www.debian.org/releases/testing/`, e `http://www.debian.org/releases/testing/installmanual` (trabalho em progresso, pode algumas vezes não existir).

Embora este capítulo tenha sido inicialmente escrito nos dias do lançamento do instalador da Potato, a maior parte dos seus conteúdos foi atualizada para o instalador do Woody e são instaladores muito similares. Como o Sarge usará um instalador totalmente novo, por favor, use isso como ponto de referência para o instalador do Sarge.

3.1 Dicas para instalação de sistemas Linux em geral

Não esqueça de conferir `http://www.debian.org/CD/netinst/` se você está procurando uma imagem de CD compacto para o instalador Debian.

Usar os sabores `testing` ou `unstable` do Debian aumenta o risco de ter bugs sérios. O risco pode ser gerenciado implementando um esquema de multi-inicialização com um sabor mais estável do Debian ou implementando o truque legal do `chroot` com o mais estável como descrito em '`chroot`' on page 131. O último permitirá que você execute diferentes sabores do Debian simultaneamente em consoles diferentes.

3.1.1 O básico sobre a compatibilidade de hardware

Linux é compatível com o hardware de muitos PC's e pode ser instalado frequentemente em qualquer sistema. Para mim foi tão fácil quanto instalar Windows 95/98/Me. E a lista de hardwares compatíveis parece continuar crescendo.

Se você tem um PC portátil (laptop), confira Linux on Laptops (`http://www.linux-laptop.net/`) para instalação dos ponteiros (pointers) por marca e modelo.

Minha recomendação sobre hardware para PC de mesa (desktop) é "Apenas seja conservador":
- SCSI ao invés de IDE para o trabalho, disco rígido IDE/ATAPI para uso privado.
- IDE/ATAPI CD-ROM (ou CD-RW).
- PCI ao invés de ISA, especialmente para placas de rede (NIC).
- Use uma placa de rede barata. Tulip para PCI, NE2000 para ISA são boas escolhas.
- Evite PCMCIA (portátil) em sua primeira instalação Linux.
- Sem teclados USB, mouse ... a menos que você queira aventurar-se.

Se você tem uma máquina lenta, tirar fora o disco rígido e colocá-lo em outra máquina mais rápida é uma boa idéia.

3.1.2 Determinando hardware e chip set de um PC

Durante a instalação, lhe será pedido que identifique o hardware ou chip set do PC. Às vezes esta informação pode não parecer fácil de encontrar. Aqui vai um método:
1 Abra o gabinete de seu PC e dê uma olhada dentro.
2 Anote os códigos de identificação dos chips grandes na placa de vídeo, placa de rede, chip próximo às portas seriais, chip próximo às portas IDE.
3 Anote o número de cartões impressos no verso das placas PCI e ISA.

3.1.3 Determinando o hardware de um PC via Debian

Os comandos a seguir devem dar alguma idéia da configuração e hardware presentes num sistema Linux.

```
$ lspci -v |pager
$ pager /proc/pci
$ pager /proc/interrupts
$ pager /proc/ioports
$ pager /proc/bus/usb/devices
```

Estes comandos podem ser rodados durante o processo de instalação num terminal virtual teclando Alt-F2.

Usos típicos de interrupções:
- IRQ0: saída do temporizador (8254)
- IRQ1: controlador de teclado
- IRQ2: cascata para IRQ8–IRQ15 no PC-AT
- IRQ3: porta serial secundária (io-port=0x2F8) (`/dev/ttyS1`)
- IRQ4: porta serial primária (io-port=0x3F8) (`/dev/ttyS0`)
- IRQ5: livre (era a interrupção do HD no XT)
- IRQ6: controlador de disquetes (io-port=0x3F0) (`/dev/fd0`, `/dev/fd1`)

- IRQ7: porta paralela (io-port=0x378) (`/dev/lp0`)
- IRQ8: rtc
- IRQ9: interrupção de software (int 0x0A), redireciona para IRQ2
- IRQ10: livre
- IRQ11: livre
- IRQ12: Mouse PS/2
- IRQ13: livre (era do co-processador matemático 80287)
- IRQ14: controlador IDE primário (`/dev/hda`, `/dev/hdb`)
- IRQ15: controlador IDE secundário (`/dev/hdc`, `/dev/hdd`)

Para dispositivos USB, as classes de dispositivos são listadas em `/proc/bus/usb/devices` como `Cls=`*`nn`*:

- Cls=00 : Sem uso
- Cls=01 : Audio (alto-falante etc.)
- Cls=02 : Comunicação (MODEM, NIC, . . .)
- Cls=03 : HID (Dispositivo de interface humana: teclado, mouse, joystick)
- Cls=07 : Impressora
- Cls=08 : Armazenamento de massa (FDD, drive CD/DVD, HDD, Flash, . . .)
- Cls=09 : Hub (hub USB)
- Cls=255 : Específico de fabricante

Se a classe de um dispositivo não for 255, o Linux tem suporte ao dispositivo.

3.1.4 Determinando o hardware de um PC por meio de outros SO's

A informação sobre o hardware pode ser obtida a partir de outros SO's:

Instale outra distribuição comercial Linux. A detecção de hardware tende, por ora, a ser melhor nestas do que é na Debian. (Esta situação deve mudar assim que o `debian-installer` for introduzido com a Sarge.

Instale Windows. A configuração do hardware pode ser obtida pressionando o botão direito do mouse em "My Computer" para obter Propriedades / Gerenciador de Dispositivos. Anote todas as informações sobre os recursos tais como IRQ, endereço de porta I/O e DMA. Algumas placas ISA antigas podem precisar ser configuradas sob o DOS e usadas em conformidade.

3.1.5 Um mito sobre o Lilo

"O Lilo está limitado a 1024 cilindros." Errado!

O `lilo` mais recente, usado após a Debian Potato tem suporte a lba32. Se o BIOS de sua placa mãe é suficientemente recente para suportar lba32, o `lilo` deve ser capaz de carregar além da limitação do cilindro 1024.

Apenas certifique-se de adicionar a linha "lba32" em algum lugar próximo ao início de seu arquivo `lilo.conf` se você tem mantido um `lilo.conf.` antigo. Confira `/usr/share/doc/lilo/Manual.txt.gz`

3.1.6 GRUB

O novo carregador de inicialização `grub` do projeto GNU HURD pode ser instalado no sistema Debian Woody:

```
# apt-get update
# apt-get install grub-doc
# mc /usr/share/doc/grub-doc/html/
... leia o conteúdo
# apt-get install grub
# pager /usr/share/doc/grub/README.Debian.gz
... leia-o :)
```

Para editar o menu do GRUB, edite `/boot/grub/menu.lst`. Veja 'Definindo parâmetros de inicialização do GRUB' on page 112 para saber como configurar os parâmetros de inicialização durante esta, pois é ligeiramente diferente da configuração do `lilo`.

3.1.7 Escolha dos disquetes de inicialização

Para Potato, gostava do conjunto de discos IDEPCI para instalação normal em um computador de mesa (desktop). Para Woody, gosto do conjunto de discos de inicialização bf2.4. Ambos usam uma versão de `boot-floppies` para criar os disquetes de inicialização.

Se você tem uma placa de rede PCMCIA, você precisa usar o conjunto padrão de discos de inicialização (o maior número de disquetes mas todos os módulos disponíveis) e configurar a NIC na configuração PCMCIA; não tente configurar uma placa NIC no diálogo de configuração padrão da rede.

Para sistemas especiais, você pode precisar criar um disco de recuperação otimizado. Isto pode ser feito substituindo a imagem do kernel denominada "linux" no disco de recuperação Debian sobrescrevendo-a com outra imagem comprimida do kernel compilada para a máquina fora do sítio (off-site). Detalhes são documentados em `readme.txt` no disco de recuperação. O disquete de recuperação usa o sistema de arquivos MS-DOS, assim você pode usar qualquer sistema para lê-lo e editá-lo. Isto deve tornar mais fácil a vida das pessoas com uma placa de rede especial, etc.

Para o Sarge, espera-se usar o `debian-installer` e/ou `pgi` para criar os disquetes de inicialização.

3.1.8 Instalação

Siga as instruções oficiais encontradas em `http://www.debian.org/releases/stable/installmanual` ou `http://www.debian.org/releases/testing/installmanual` (trabalho em progresso; pode, às vezes, não existir).

Se você está instalando um sistema usando os `boot-floppies` da distribuição `testing`, você pode precisar abrir um terminal durante a instalação pressionando Alt-F2 e edite manualmente as entradas de `/etc/apt/sources.list`, substituindo "stable" por "testing" para ajustar as fontes do APT.

Costumo instalar o `lilo` em lugares como `/dev/hda3`, e instalo `mbr` em `/dev/hda`. Isto diminui o risco de sobrescrever a informação de inicialização.

Aqui está o que escolho durante o processo de instalação.

- MD5 passwords "yes"
- shadow passwords "yes"
- Install "advanced" (dselect **) e selecione
  - Exclua emacs (se selecionado), nvi, tex, telnet, talk(d);
  - Inclua mc, vim e nano-tiny ou elvis-tiny. Veja '`dselect`' on page 83. Mesmo que você seja fã do Emacs, evite-o por enquanto e se contente com nano durante a instalação. Evite também instalar grandes pacotes tais como TeX (Potato fazia isto) nesta fase. Veja 'Editores de recuperação' on page 212 para entender as razões que justificam instalar nano-tiny ou elvis-tiny aqui.
- Para todas as perguntas de configuração durante o diálogo de instalação de cada pacote: "y" (substitua versão atual).
- `exim`: seleciono 2 para a máquina pois envio correio eletrônico através do servidor SMTP de meu provedor de internet.

Para mais informações sobre dselect, veja '`dselect`' on page 83.

3.1.9 Hosts e IP a usar para LAN

Exemplo de configuração de uma LAN (C subnet: 192.168.1.0/24):

```
Internet
   |
   +--- ISP externo fornece serviço POP (acessado pelo fetchmail)
   |
   ISP de ponto de acesso fornece serviço DHCP e retransmissão SMTP
   |                     :
  Cable modem          (Dialup)
   |                     :
Porta externa da máquina gateway da LAN: eth0 (IP dado pelo DHCP do
ISP)
 use PC portátil antigo (IBM Thinkpad, 486 DX2 50 MHz, 20MB RAM)
 rode kernel Linux 2.4 com sistema de arquivos ext3.
 rode o pacote "ipmasq" (com as correções (patch) mais fortes, NAT e
 firewall)
 rode o pacote "dhcp-client" configurado para eth0 (sobrescreve a
 configuração de DNS)
 rode o pacote "dhcp" configurado para eth1
```

```
  rode "exim" como e modo de host auxiliar (smarthost, mode 2)
  rode "fetchmail" em longos períodos (fallback)
  rode "bind" como o cache do servidor de nomes para a Internet a partir
              da LAN
              como servidor de nomes autoritativo para o domínio da LAN a
              partir da LAN
  rode "ssh" na porta 22 e 8080 (conecte de qualquer lugar)
  rode "squid" como o servido de cache para o arquivo Debian (para APT)
Porta interna da máquina gateway da LAN: eth1 (IP = 192.168.1.1, fixo)
                        |
         +--- LAN Switch (100base T) ---+
         |                              |
Alguns clientes de IP fixos na LAN     Alguns clientes DHCP na LAN
(IP = 192.168.1.2-127, fixo)    (IP = 192.168.1.128-200, dinâmico)
```

Veja 'Configuração de rede' on page 183 para detalhes de configuração da rede. Veja 'Construindo um roteador gateway' on page 207 para detalhes de configuração do servidor gateway da LAN.

3.1.10 Contas de usuário

Para manter a consistência entre diferentes máquinas, as primeiras contas são sempre as mesmas em meu sistema.

Sempre crio a primeira conta de usuário com um nome como "admin" (uid=1000). Eu reenvio todo email para o root para lá. Esta conta está associada ao grupo `adm` (veja "'Por que o GNU `su` não suporta o grupo `wheel`"' on page 140), para o qual pode ser dado um grande número de privilégios root através do programa `su` usando PAM ou o comando `sudo`. Veja 'Adicionar uma conta de usuário' on page 49 para detalhes.

3.1.11 Criando sistemas de arquivo

Partição do disco rígido

Eu prefiro usar partições diferentes em diferentes árvores de diretório para limitar o estrago numa eventual quebra do sistema. Por exemplo,

```
/          == (/ + /boot + /bin + /sbin)
           == 50MB+
/tmp       == 100MB+
/var       == 100MB+
/home      == 100MB+
/usr       == 700MB+ com X
/usr/local == 100MB
```

O tamanho do diretório `/usr` depende muito das aplicações do sistema X Window e da documentação. `/usr/` pode ter 300MB se rodamos apenas um terminal console, enquanto 2GB–3GB não é um tamanho incomum se temos muitas aplicações Gnome instaladas. Quando `/usr` ficar muito grande, mover `/usr/share/` para uma partição diferente é o remédio mais efetivo. Com os novos grandes kernels Linux 2.4 pré-empacotados, pode ser necessário mais do que 200MB.

Por exemplo, o estado atual de minha máquina gateway para Internet é o seguinte (saída do comando `df -h`):

```
Filesystem            Size  Used Avail Use% Mounted on
/dev/hda3             300M  106M  179M  38% /
/dev/hda7             100M   12M   82M  13% /home
/dev/hda8             596M   53M  513M  10% /var
/dev/hda6             100M  834k   94M   1% /var/lib/cvs
/dev/hda9             596M  222M  343M  40% /usr
/dev/hda10            596M  130M  436M  23% /var/cache/apt/archives
/dev/hda11            1.5G  204M  1.2G  14% /var/spool/squid
```

(A grande área reservada para `/var/spool/squid/` é para um proxy cache dos pacotes baixados.)

Segue abaixo a saída de `fdisk -l` que fornece uma idéia da estrutura de particionamento:

```
# fdisk -l /dev/hda # comentário

/dev/hda1            1        41    309928+   6  FAT16 # DOS
/dev/hda2           42        84    325080   83  Linux # (não usada)
/dev/hda3   *       85       126    317520   83  Linux # Principal
/dev/hda4          127       629   3802680    5  Extended
/dev/hda5          127       143    128488+  82  Linux swap
/dev/hda6          144       157    105808+  83  Linux
/dev/hda7          158       171    105808+  83  Linux
/dev/hda8          172       253    619888+  83  Linux
/dev/hda9          254       335    619888+  83  Linux
/dev/hda10         336       417    619888+  83  Linux
/dev/hda11         418       629   1602688+  83  Linux
```

Há poucas partições não usadas. São para instalação de uma segunda distribuição Linux ou para expandir árvores de diretórios crescentes.

Montar sistema de arquivos

A montagem do sistema de arquivos acima é realizada corretamente com o seguinte `/etc/fstab`:

```
   # /etc/fstab: informação de sistema de arquivos estático
   #
   # file system   mount point     type    options                 dump pass
   /dev/hda3       /               ext2    defaults,errors=remount-ro 0 1
   /dev/hda5       none            swap    sw                       0 0
   proc            /proc           proc    defaults                 0 0
   /dev/fd0        /floppy         auto    defaults,user,noauto     0 0
   /dev/cdrom      /cdrom          iso9660 defaults,ro,user,noauto 0 0
   #
   # manter partições separadas
   /dev/hda7       /home           ext2    defaults                 0 2
   /dev/hda8       /var            ext2    defaults                 0 2
   /dev/hda6       /var/lib/cvs    ext2    defaults                 0 2
   # noatime aumentará velocidade de acesso a arquivo para leitura
   /dev/hda9       /usr            ext2    defaults,noatime         0 2
   /dev/hda10      /var/cache/apt/archives ext2    defaults         0 2

   # partição muito grande para cache do proxy
   /dev/hda11      /var/spool/squid ext2   rw                       0 2

   # DOS inicializável de segurança (backup)
   /dev/hda1       /mnt/dos        vfat    rw,noauto                0 0
   # sistema Linux inicializável de segurança (backup) (não feito)
   /dev/hda2       /mnt/linux      ext2    rw,noauto                0 0
   #
   # montagens nfs
   mickey:/        /mnt/mickey     nfs     ro,noauto,intr           0 0
   goofy:/         /mnt/goofy      nfs     ro,noauto,intr           0 0
   # minnie:/ /mnt/minnie smbfs ro,soft,intr,credentials={filename} 0 2
```

Para NFS, uso `noauto,intr` combinada com a opção padrão `hard`. Deste modo, é possível safar-se de um processo congelado devido a uma conexão morta usando Ctrl-C.

Para uma máquina Windows conectada com Samba (smbfs), `rw,auto,soft,intr` pode ser uma boa idéia. Veja 'Configuração do Samba' on page 40.

Para os disquetes, `noauto,rw,sync,user,exec` ao contrário previne contra a corrupção de arquivos após a ejeção acidental do disquete antes de desmontá-lo, mas isto retarda o processo de escrita.

Montagem autofs

Pontos chave para a montagem automática:

- Carregue o módulo `vfat` para permitir que `/etc/auto.misc` contenha `-fstype=auto`:

```
# modprobe vfat # antes da tentativa de acesso ao disquete
 ... ou, para automatizar esta configuração,
# echo "vfat" >> /etc/modules
 ... e reinicialize o sistema.
```

- Configure `/etc/auto.misc` como se segue:

```
floppy -fstype=auto,sync,nodev,nosuid,gid=100,umask=000 :/dev/fd0
 ... onde gid=100 é "users".
```

- Crie os links `cdrom` e `floppy` em `/home/`*`usuario`*, que apontem para `/var/autofs/misc/cdrom` e `/var/autofs/misc/floppy` respectivamente.
- Adicione o *usuario* um membro do grupo “users”.

Montagem NFS

O servidor NFS Linux (goofy) fica atrás de um firewall (gateway). Eu tenho uma política de segurança bastante aberta em minha LAN já que sou eu o único usuário. Para permitir acesso NFS, o lado do servidor precisa adicionar a `/etc/exports` o seguinte:

```
# /etc/exports: the access control list for file systems which may be
#               exported to NFS clients.  See exports(5).
/       (rw,no_root_squash)
```

Isto é necessário para ativar o servidor NFS, além de instalar e ativar os pacotes de servidor e cliente NFS.

Por simplicidade, normalmente crio uma partição única de 2GB para uma instalação experimental ou secundária de Linux. Opcionalmente, compartilho a partição de troca (swap) e a partição `/tmp` para estas instalações. Um esquema multipartição também é usado nestes casos. Se é necessário apenas um simples sistema console, 500MB pode ser mais do que suficiente.

3.1.12 Guia sobre memória DRAM

O que se segue é um guia grosseiro sobre DRAM.

```
  4MB:  Mínimo para o kernel Linux funcionar.
 16MB:  Mínimo para um sistema console razoável.
 32MB:  Mínimo para um sistema X simples.
 64MB:  Mínimo para o sistema X com GNOME/KDE.
128MB:  Confortável para o sistema X com GNOME/KDE.
256MB:  Por que não se você pode se dar ao luxo disto? DRAM é barata.
```

Usando a opção de inicialização `mem=4m` (ou lilo `append=“mem=4m”`) mostrará como o sistema funcionaria com 4MB de memória instalada. É necessário o parâmetro de inicialização do lilo para sistemas que contém mais do que 64 MB de memória e um BIOS antigo.

3.1.13 Espaço de troca (swap)

Eu uso o seguinte guia para o espaço de troca:

- Cada partição de troca é < 128MB (antigo kernel 2.0), < 2GB (em kernels recentes)
- Total = ou 1 a 2 vezes a RAM instalada ou de 128MB a 2GB como orientação
- Espalhe-as nos diferentes discos e monte-as com as opções `sw,pri=1` em `/etc/fstab`. Isto garante que o kernel faça um striping RAID das partições de troca (swap) e ofereça o máximo de performance na troca.
- Use uma porção central do disco rígido, quando possível.

Mesmo que você nunca precise disto, algum espaço de troca (128MB) é desejável pois assim o sistema ficará lento antes de congelar com um programa que consuma a memória.

3.2 Configuração do bash

Eu modifico os scripts shell de inicialização a meu gosto no sistema:

```
/etc/bash.bashrc        Utilize sua cópia privada
/etc/profile            Mantenha a cópia da distribuição ( \w -> \W)
/etc/skel/.bashrc       Utilize sua cópia privada
/etc/skel/.profile      Utilize sua cópia privada
/etc/skel/.bash_profile Utilize sua cópia privada
~/.bashrc               Utilize sua cópia privada para todas as contas
~/.profile              Utilize sua cópia privada para todas as contas
~/.bash_profile         Utilize sua cópia privada para todas as contas
```

Veja detalhes nos meus scripts de exemplo (`http://www.debian.org/doc/manuals/debian-reference/examples/`). Gosto de um sistema transparente, por isto configuro `umask` para 002 ou 022.

`PATH` é configurado pelos seguintes arquivos de configuração, nesta ordem:

```
/etc/login.defs  - antes o shell configura PATH
/etc/profile     (pode chamar /etc/bash.bashrc)
~/.bash_profile  (pode chamar ~/.bashrc)
```

3.3 Configuração do mouse

3.3.1 Mouses PS/2

No caso de um mouse com conector do tipo PS/2 em uma placa mãe ATX, o fluxo do sinal deveria ser:

```
mouse -> /dev/psaux -> gpm -> /dev/gpmdata = /dev/mouse -> X
```

Aqui, um link simbólico para `/dev/mouse` é criado e aponta para `/dev/gpmdata` para tornar alguns utilitários de configuração felizes e mais a reconfiguração mais fácil. (Por exemplo, se você decide não usar o daemon `gpm`, apenas aponte o link simbólico `/dev/mouse` para `/dev/psaux` após livrar-se do daemon `gpm`.)

Este fluxo de sinal permite que o teclado e o mouse sejam desplugados e reinicializados com o `gpm` sendo rodado novamente na reconexão. O X permanecerá vivo!

O protocolo do fluxo de sinal entre a saída do `gpm` e a entrada do X pode ser implementado de dois modos, mutuamente exclusivos, como "ms3" (usa o protocolo Microsoft 3-button serial mouse) ou "raw" (usa o mesmo protocolo do mouse que está conectado), e esta escolha determina a escolha do protocolo usado na configuração do X.

Abaixo, mostrarei os exemplos de configuração usando um mouse Logitech 3-button (mouse Unix-style tradicional) PS/2 como modelo.

Se você é um dos desafortunados cuja placa de vídeo não é suportada pelo novo X4 e precisa usar o antigo X3 (algumas placas ATI 64 bit), configure `/etc/X11/XF86Config` ao invés de `/etc/X11/XF86Config-4` nos exemplos a seguir e instale os pacotes X3.

A perspectiva do protocolo ms3

```
/etc/gpm.conf              | /etc/X11/XF86Config-4
==========================+======================================
device=/dev/psaux          | Section "InputDevice"
responsiveness=            |  Identifier "Configured Mouse"
repeat_type=ms3            |  Driver     "mouse"
type=autops2               |  Option     "CorePointer"
append=""                  |  Option     "Device"   "/dev/mouse"
sample_rate=               |  Option     "Protocol" "IntelliMouse"
                           | EndSection
```

Se esta perspectiva é usada, o ajuste do tipo de mouse é feito apenas editando `gpm.conf` e a configuração do X permanece a mesma. Veja meus scripts de exemplo (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

A perspectiva do protocolo raw

```
/etc/gpm.conf              | /etc/X11/XF86Config-4
==========================+======================================
device=/dev/psaux          | Section "InputDevice"
responsiveness=            |  Identifier "Configured Mouse"
repeat_type=raw            |  Driver     "mouse"
type=autops2               |  Option     "CorePointer"
```

```
append=""                  |  Option     "Device"   "/dev/mouse"
sample_rate=               |  Option     "Protocol" "MouseManPlusPS/2"
                           | EndSection
```

Se esta perspectiva é usada, o ajuste do tipo de mouse é feito editando `gpm.conf` assim como ajustando a configuração do X.

Como ajustar um mice diferente

O tipo de dispositivo do `gpm autops2` provavelmente autodetecta muitos dos mouses PS/2 no mercado. Infelizmente, nem sempre funciona e não está disponível nas versões pré-Woody. Para tais casos, tente usar `ps2`, ou `imps2` em `gpm.conf` ao invés de `autops2`. Para descobrir quais tipos de mouse o `gpm` reconhece: `gpm -t help`. Veja `gpm(8)`.

Se um mouse PS/2 de 2 botões é usado, configure o X para habilitar `Emulate3Buttons`. A diferença de protocolo entre o mouse de 2 botões e o mouse de 3 botões é autodetectada a auto-ajustada para o `gpm` após apertar o botão central uma vez.

Para o protocolo X com 'A perspectiva do protocolo raw' on the preceding page ou com `gpm`, use:

- `IntelliMouse`: mouse serial (repetidor `gpm` com "ms3")
- `PS/2`: mouse PS/2 (sempre teste este primeiro)
- `IMPS/2`: mouses PS/2 (2, 3, ou mouses de rolagem, melhor)
- `MouseManPlusPS/2`: mouse Logitech PS/2
- `...`

Veja mais em Suporte a Mouse no XFree86 (`http://www.xfree86.org/current/mouse.html`).

Sab-se que mouses de rolagem Microsoft típicos funcionam melhor com:

```
/etc/gpm.conf              | /etc/X11/XF86Config-4
=========================+=====================================
device=/dev/psaux          | Section "InputDevice"
responsiveness=            |  Identifier "Configured Mouse"
repeat_type=raw            |  Driver     "mouse"
type=autops2               |  Option     "CorePointer"
append=""                  |  Option     "Device"   "/dev/mouse"
sample_rate=               |  Option     "Protocol" "IMPS/2"
                           |  Option     "Buttons" "5"
                           |  Option  "ZAxisMapping" "4 5"
                           | EndSection
```

Para alguns pequenos portáteis Toshiba novos, ativar `gpm` antes do PCMCIA no script de inicialização System-V pode ajudar a prevenir o travamento do sistema. Estranho mas verdadeiro.

3.3.2 Mouses USB

Certifique-se que você tem todas as funções do kernel necessárias ativadas através de configurações feitas durante a compilação do kernel ou em módulos:

- Em “Input core support”:
  - “Input core support” (CONFIG_INPUT, `input.o`),
  - “Mouse support” (CONFIG_INPUT_MOUSEDEV, `mousedev.o`),
- Em “USB support”:
  - “Support for USB” (CONFIG_USB, `usbcore.o`),
  - “Preliminary USB device filesystem” (CONFIG_USB_DEVICEFS),
  - “UHCI” or “OHCI” (CONFIG_USB_UHCI || CONFIG_USB_UHCI_ALT || CONFIG_USB_OHCI, `usb-uhci.o` || `uhci.o` || `usb-ohci.o`),
  - “USB Human Interface Device (full HID) support” (CONFIG_USB_HID, `hid.o`), e
  - “HID input layer support” (CONFIG_USB_HIDINPUT)

Aqui, nomes em minúsculas são nomes de módulos.

Se você não está usando dvfs, crie um dispositivo nó (node) `/dev/input/mice` com major 13 e minor 63 como se segue:

```
# cd /dev
# mkdir input
# mknod input/mice c 13 63
```

Para mouses **USB** com roda típicos, as combinações de configuração deveriam ser:

```
/etc/gpm.conf           | /etc/X11/XF86Config-4
========================+=====================================
device=/dev/input/mice  | Section "InputDevice"
responsiveness=         |  Identifier "Generic Mouse"
repeat_type=raw         |  Driver     "mouse"
type=autops2            |  Option     "SendCoreEvents" "true"
append=""               |  Option     "Device"   "/dev/input/mice"
sample_rate=            |  Option     "Protocol" "IMPS/2"
                        |  Option     "Buttons" "5"
                        |  Option  "ZAxisMapping" "4 5"
                        | EndSection
```

Veja o Linux USB Project (`http://www.linux-usb.org/`) para mais informações.

3.3.3 Touch pad

Apesar de o touch pad no computador laptop emular um mouse de 2 botões PS/2 como seu comportamento padrão, o pacote `tpconfig` possibilita controle total do dispositivo. Por exemplo, configurando `OPTIONS="--tapmode=0"` no arquivo `/etc/default/tpconfig`, desabilita-se o incômodo comportamento de “clique por batida leve”. Configure o arquivo `/etc/gpm.conf` como a seguir para usar tanto o touch pad como o mouse USB externo no console:

```
device=/dev/psaux
responsiveness=
repeat_type=ms3
type=autops2
append="-M -m /dev/input/mice -t autops2"
sample_rate=
```

3.4 Configuração de NFS

Habilite NFS configurando `/etc/exports`.

```
# echo "/ *.domainname-for-lan-hosts(rw,no_root_squash,nohide)" \
        >> /etc/exports
```

Veja meus scripts de exemplo para detalhes (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

3.5 Configuração do Samba

Referências:

- `http://www.samba.org/`
- `samba-doc` package

Configurar o Samba com modo “share” é muito mais fácil pois este cria discos compartilhados tipo WfW. Mas é preferível configurá-lo com mode “ser”.

O Samba pode ser configurado através de `debconf` ou `vi`:

```
# dpkg-reconfigure --priority=low samba # em Woody
# vi /etc/samba/smb.conf
```

Veja meus scripts de exemplo para detalhes (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

Pode-se adicionar um novo usuário ao arquivo `smbpasswd` via `smbpasswd`:

```
$su -c "smbpasswd -a username"
```

Certifique-se de usar senhas criptografadas para uma compatibilidade excelente.

Configure `os level` de acordo com o seguinte sistema de equivalências (maior o número, maior a prioridade como servidor):

```
0:      Samba com uma atitude folgada (nunca se tornará um navegador
mestre)
1:      Wfw 3.1, Win95, Win98, Win/me?
16:     Win NT WS 3.51
17:     Win NT WS 4.0
32:     Win NT SVR 3.51
33:     Win NT SVR 4.0
255:    Samba com enorme poder
```

Certifique-se de que usuários sejam membros do grupo dono do diretório que deu acesso compartilhado e que o caminho do diretório tenha seu bit de execução configurado para acesso.

3.6 Configuração da Impressora

O método tradicional é `lpr/lpd`. Há um novo sistema: CUPS™ (Common UNIX Printing System). PDQ é outra opção. Veja Linux Printing HOWTO (`http://www.tldp.org/HOWTO/Printing-HOWTO.html`) para mais informações.

3.6.1 `lpr/lpd`

Para os spoolers do tipo `lpr/lpd` (`lpr`, `lprng`, e `gnulpr`), configure `/etc/printcap` como se segue, deste que conectado a um impressora PostScript ou somente texto (o básico):

```
lp|alias:\
        :sd=/var/spool/lpd/lp:\
        :mx#0:\
        :sh:\
        :lp=/dev/lp0:
```

Significado das linhas acima:

- Linha cabeçalho: *lp* – nome do spool, *alias* = alias
- mx#0 – tamanho máximo do arquivo é ilimitado
- sh – suprime a impressão da primeira página do cabeçalho
- lp=/dev/lp0 – dispositivo de impressora local, ou porta@host, se remota

Esta é uma boa configuração se você está conectado a uma impressora PostScript. Além disto, quando imprimir a partir de uma máquina Windows através do Samba, esta é uma boa configuração para qualquer impressora suportada por Windows (nenhuma comunicação bidirecional é suportada). Você tem d e selecionar a configuração da impressora correspondente na máquina Windows.

Se você não tem uma impressora PostScript, você precisa configurar um sistema de filtros usando `gs`. Há várias ferramentas para autoconfiguração com a finalidade de configurar `/etc/printcap`. Quaisquer destas combinações é uma opção:

- `gnulpr`, (`lpr-ppd`) e `printtool` —eu uso isto.
- `lpr` e `apsfilter`
- `lpr` e `magicfilter`
- `lprng` e `lprngtool`
- `lprng` e `apsfilter`
- `lprng` e `magicfilter`

Para rodar uma ferramenta de configuração GUI, tal como `printtool`, veja 'Obtendo root no X' on page 158 para obter privilégios de root. Impressoras spools criadas com `printtool` usam `gs` e agem como impressoras PostScript. Assim, quando as acessar, use drivers de impressora PostScript. Na perspectiva do Windows, "Apple LaserWriter" é o único padrão.

3.6.2 CUPS™

Instale o Sistema de Impressão Comum UNIX (Common UNIX Printing System) (ou CUPS™): # apt-get install cupsys cupsys-bsd cupsys-client cupsys-driver-gimpprint # apt-get install foomatic-db-engine foomatic-db-hpijs # apt-get install foomatic-filters-ppds foomatic-gui

Então configure o sistema usando qualquer navegador web:

```
$ meunavegador http://localhost:631
```

Por exemplo, para adicionar sua impressora em alguma porta à lista de impressoras acessíveis:

- clique "Printers" na página principal e, e então "Add Printer",
- tecle "root" para o nome de usuário e sua senha,
- adicione a impressora seguindo os prompts,
- volte para a página "Printers" e clique "Configure Printer", e
- configure o tamanho do papel, resolução, e outros parâmetros.

Mais informação em `http://localhost:631/documentation.html` e `http://www.cups.org/cups-help.html`.

3.7 Outras dicas de instalação no host

3.7.1 Instalar mais alguns pacotes após a instalação inicial

Desde que você já o tenha feito, já terá um pequeno mas funcional sistema Debian. É um bom momento para instalar pacotes maiores.

- Rode `tasksel`. Veja 'Instalar *tarefa* com o `tasksel` ou `aptitude`' on page 82.

  Você pode escolher estes, se necessário:
  - End-user (usuário final) – X Window System
  - Development (desenvolvimento) – C and C++
  - Development – Python
  - Development – Tcl/Tk
  - Miscellaneous – TeX/LaTeX environment
  - Para outros, eu prefiro usar o `tasksel` como um guia, olhando os componentes listados sob <Task Info> e instalando-os seletivamente através do `dselect`.
- Rode `dselect`.

  Aqui, a primeira coisa que talvez você queira é selecionar seu editor favorito e quaisquer programas que você necessitar. Você pode instalar muitas variações do Emacs ao mesmo tempo. Veja '`dselect`' on page 83 e 'Editores populares' on page 211.

  Você pode também substituir alguns dos pacotes padrões por outros com muitos recursos.
  - ...
- ...

Normalmente, eu edito `/etc/inittab` para desligar facilmente.

```
...
# O que fazer quando CTRL-ALT-DEL é pressionado.
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -h now
...
```

3.7.2 Módulos

Módulos para os gerenciadores de dispositivo são configurados durante a instalação inicial. Posteriormente `modconf` oferece um modo de configuração dos módulos que é gerenciado por menus. Este programa é muito útil quando alguns módulos foram esquecidos na instalação inicial ou um novo kernel foi instalado após esta.

O nome dos módulos pré-carregados devem ser listados em `/etc/modules`. Eu também uso `lsmod` e `depmod` para controlá-los manualmente.

Certifique-se, também de adicionar algumas linhas em `/etc/modules` para lidar com mascaramento de IP (FTP, etc.) em kernels 2.4. Veja 'O kernel 2.4 modularizado' on page 103, especificamente 'Função de rede' on page 105.

3.7.3 Configuração básica de CD-RW

Edite os seguintes arquivos:

```
/etc/lilo.conf  (adicione append="hdc=ide-scsi ignore=hdc",
                 rode lilo para ativar)
/dev/cdrom      (symlink # cd /dev; ln -sf scd0 cdrom)
/etc/modules    (adicione "ide-scsi" e "sg". Se necessário, "sr" após
                isto.)
```

Veja 'Gravadores de CD' on page 143 para detalhes.

3.7.4 Memória grande e autodesligamento

Edite o `/etc/lilo.conf` como se segue para configurar os parâmetros de inicialização para memória grande (para kernels 2.2) e autodesligamento (para APM):

```
append="mem=128M apm=on apm=power-off noapic"
```

Rode o `lilo` para instalar estas configurações. `apm=power-off` é necessário para um kernel SMP e `noapic`, para evitar problemas com meu hardware SMP cheio de bugs. O mesmo pode ser feito diretamente, inserindo as opções no prompt de inicialização. Veja 'Outros truques de inicialização com o prompt de inicialização' on page 112.

Se o APM está compilado como um módulo, como nos kernels 2.4 padrão da Debian, rode `insmod apm power_off=1` após a inicialização ou configure `/etc/modules` com:

```
# echo "apm power_off=1" >>/etc/modules
```

Alternativamente, ao compilar suporte a ACPI obtém-se o mesmo resultado com kernels mais novos e parece ser mais SMP-amigável (isto requer uma placa mãe mais nova). O kernel 2.4 em placas mãe mais novas detecta corretamente grandes memórias.

```
CONFIG_PM=y
CONFIG_ACPI=y
...
CONFIG_ACPI_BUSMGR=m
CONFIG_ACPI_SYS=m
```

e adicione as seguintes linhas em `/etc/modules` nesta ordem:

```
ospm_busmgr
ospm_system
```

Ou recompile o kernel com todas as opções acima configuradas como "y". De qualquer maneira, nenhum destes parâmetros de inicialização são necessários com ACPI.

3.7.5 Estranhos problemas de acesso a alguns websites

Kernels linux mais novos habilitam ECN por padrão (default), o que pode causar problemas de acesso em alguns websites com roteadores ruins. Para checar o status ECN:

```
# cat /proc/sys/net/ipv4/tcp_ecn
 ... ou
# sysctl net.ipv4.tcp_ecn
```

Para desabilitá-lo, use:

```
# echo "0" > /proc/sys/net/ipv4/tcp_ecn
 ... ou
# sysctl -w net.ipv4.tcp_ecn=0
```

Para desabilitar TCP ECN em toda inicialização, edite `/etc/sysctl.conf` e adicione:

```
net.ipv4.tcp_ecn = 0
```

3.7.6 Configuração Dialup PPP

Instale o pacote `pppconfig` para configurar o acesso dial-up PPP.

```
# apt-get install pppconfig
# pppconfig
 ... siga as instruções para configurar dialup PPP
# adduser nome_usuario dip
 ... permita que nome_usuario acesse dialup PPP
```

Acesso dialup PPP pode ser iniciado pelo usuário (*nome_usuario*):

```
$ pon nome_ISP  # inicia o acesso PPP a seu ISP
 ... navegue na Internet
$ poff nome_ISP # encerra o acesso PPP, nome_ISP opcional
```

Veja `/usr/share/doc/ppp/README.Debian.gz` para mais detalhes.

Alternativamente, o pacote `wvdial` pode ser usado para configurar o acesso dialup PPP. Por favor, note que há um bug `http://bugs.debian.org/82095` bem conhecido, que algumas vezes impede que usuários não-root façam a discagem (dialing).

Todos os discadores (dialers) usam o daemon `pppd`, que inicia os programas encontrados em `/etc/ppp/ip-up.d/` ou `/etc/ppp/ip-down.d` depois de conectar ou desconectar. Isso é usado para receber e enviar mensagens.

3.7.7 Outros arquivos de configuração para fuçar em `/etc/`

Você pode querer adicionar um arquivo `/etc/cron.deny` que não venha na instalação da distribuição Debian padrão (você pode copiar `/etc/at.deny`).

Capítulo 4

Tutoriais do Debian

Essa seção fornece uma orientação básica para o mundo Debian para o verdadeiro novato. Se você tiver usado algum sistema semelhante ao Unix por algum tempo, provavelmente sabe tudo que expliquei aqui. Por favor, use isso como uma verificação da realidade.

4.1 Começando

Depois da instalação do sistema Debian em seu PC, você precisa aprender algumas coisas para torná-lo útil. Vamos dar-lhe um treinamento expresso.

4.1.1 Login em um prompt shell como root

Após reiniciar o sistema, será apresentado a você uma tela de login gráfica ou baseada em caracteres dependendo de sua seleção inicial de pacotes. Para simplificar, se lhe for apresentada uma tela de login gráfica, pressione Ctrl-Alt-F1 [1] para obter a tela de login baseada em caracteres.

Suponha que o nome de seu sistema é *`foo`*, o prompt de login parece assim:

```
foo login:
```

Digite `root`, pressione a tecla Enter e digite a senha que você selecionou durante o processo de instalação. No sistema Debian, seguindo a tradição Unix, a senha é sensível a maiúsculas/minúsculas. Então o sistema começa com uma mensagem de saudação e lhe apresenta o prompt de comando root aguardando sua entrada. [2]

```
foo login: root
```

---

[1] A tecla Ctrl esquerda, a tecla Alt esquerda e a tecla F1 pressionadas juntas.

[2] Note que se você editou a mensagem de saudação no arquivo `/etc/motd`, ela será diferente.

```
Password:
Last login: Sun Oct 26 19:04:09 2003 on tty3
Linux foo 2.4.22-1-686 #6 Sat Oct 4 14:09:08 EST 2003 i686 GNU/Linux

Most of the programs included with the Debian GNU/Linux system are
freely redistributable; the exact distribution terms for each program
are described in the individual files in /usr/share/doc/*/copyright

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

root@foo:root#
```

Você está pronto para fazer a administração do sistema a partir desse prompt de comando root. Essa conta root também é chamada de super-usuário ou usuário privilegiado. A partir dessa conta, você pode fazer qualquer coisa:

- ler, gravar, e remover quaisquer arquivos no sistema independentemente das permissões de arquivo
- ajustar as propriedades e permissões de quaisquer arquivos no sistema
- ajustar a senha de qualquer usuário não-privilegiado no sistema
- fazer login para quaisquer contas sem suas senhas

É uma idéia extremamente ruim compartilhar o acesso para a conta root compartilhando a senha. O uso de um programa como o `sudo(8)` é a maneira certa de compartilhar privilégios administrativos.

Por favor note que considera-se um bom hábito de Unix fazer login para a conta de usuário não privilegiado mesmo quando você planeja fazer atividades administrativas. Use os comandos `sudo`, `super`, ou `su -c` para obter o privilégio de root limitado quando necessário. Veja 'Trabalhando de forma mais segura – `sudo`' on page 141. [3]

4.1.2 Configurar um ambiente mínimo para novato

Eu acho que aprender um sistema de computador é como aprender um idioma estrangeiro novo. Apesar de livros tutoriais ajudarem, você tem que praticá-lo com ferramentas de ajuda. Nesse contexto, acho que é uma boa idéia instalar alguns pacotes adicionais tais como `mc`, `vim`, `lynx`, `doc-linux-text` e `debian-policy`. [4]

```
# apt-get update
 ...
# apt-get install mc vim lynx doc-linux-text debian-policy
 ...
```

---

[3] Eu tenho que admitir que usei a conta do super-usuário mais freqüentemente que o necessário apenas porque era mais fácil e eu era descuidado.

[4] Também pode ser uma boa idéia instalar `gpm`, `emacs21` e `doc-linux-html`. Veja 'Configuração do mouse' on page 36 e 'Editores' on page 211.

Se você já tiver esses pacotes instalados, nada será instalado.

4.1.3 Adicionar uma conta de usuário

Durante a instalação, normalmente se cria uma conta de usuário não privilegiado que recebe e-mails destinados à conta root. [5] Já que você não quer usar essa conta de usuário especial para as atividades de treinamento seguintes, você deve criar outra conta de usuário nova.

Supondo que você deseje que esse novo nome de usuário seja *`pinguim`*, digitar:

```
root@foo:root# adduser pinguim
... responda todas as questões
```

irá criá-la. [6] Antes de ir adiante, vamos aprender algumas coisas primeiro.

4.1.4 Alternar entre consoles virtuais

No sistema Debian padrão, há seis pseudo-terminais independentes disponíveis, isto é, você pode usar a tela de console de caracter VGA do PC como 6 terminais VT-100 comutáveis. Mude de um para outro pressionando a telcla Alt esquerdo e uma das teclas F1–F6 simultaneamente. Cada pseudo-terminal permite login independente para contas. O ambiente multi-usuário é uma grande característica do Unix, e muito importante.

Se você acidentalmente pressionou Alt-F7 em um sistema executando o Sistema X Window e a tela de console mostrar uma tela gráfica, obtenha o acesso ao console de caracter novamente pressionando Ctrl-Alt-F1. Tente mudar para um console diferente e volte ao original para exercitar isso.

4.1.5 Como desligar

Como quaisquer outros SOs modernos onde a operação de arquivos envolve cache de dados na memória, o sistema Debian precisa de um procedimento de desligamento apropriado antes que a energia possa ser desligada de forma segura para manter a integridade dos arquivos. Use o comando seguinte a partir do prompt root para desligar o sistema:

```
# shutdown -h now
```

Isso é para o modo multi-usuário normal. Se você estiver no modo usuário único, use o seguinte a partir do prompt de comando root:

---

[5] Eu tendo a nomear essa conta criada durante a instalação como `admin` mas pode ser qualquer nome arbitrário.

[6] Você pode querer adicionar esse usuário *`pinguim`* ao grupo `adm` para habilitar acesso de leitura aos muitos arquivos de log em `/var/log/`. Veja as páginas de manual `passwd(5)`, `group(5)`, `shadow(5)`, `vipw(8)`, e `vigr(8)`. Para obter os significados oficiais de usuários de grupos, veja uma versão recente do documento Usuários e Grupos (`/usr/share/doc/base-passwd/users-and-groups.html`).

```
# poweroff -i -f
```

Alternativamente, você pode pressionar Ctrl-Alt-Delete para desligar. [7]

Espere até que o sistema mostre "System halted" e então desligue a energia. Se a função APM ou ACPI tiver sido ativada pelo Linux e BIOS apropriadamente, o sistema se desligará sozinho. Veja 'Memória grande e autodesligamento' on page 44 para maiores detalhes.

4.1.6 Hora do jogo

Agora você está pronto para brincar com seu sistema Debian sem riscos desde que você use essa conta de usuário não privilegiado *`pinguim`*. [8]

Vamos fazer login para o *`pinguim`*. Se você estiver no prompt shell root, pressione Ctrl-D [9] no prompt de comando root para fechar a atividade shell root e retornar ao prompt de login. Entre seu nome de usuário recém criado *`pinguim`* e a senha correspondente. [10] Será apresentado a você o seguinte prompt de comando.

```
pinguim@foo:pinguim$
```

A partir daqui, o exemplo dado usará o prompt de comando simplificado por facilidade. Eu usarei:

- `#` : prompt shell root
- `$` : prompt shell de usuário não privilegiado

Nós começaremos aprendendo o sistema Debian primeiro da maneira fácil, 'Midnight Commander (MC)' on this page e mais tarde da maneira apropriada, 'Ambiente de trabalho semelhante ao Unix' on page 53.

4.2 Midnight Commander (MC)

O Midnight Commander (MC) é um "canivete suíço" do GNU para o console Linux e outros ambientes de terminal. Isso dá ao novato uma experiência com o console baseada em menu que é muito mais fácil de aprender que os comandos Unix padrões.

---

[7] A tecla Ctrl esquerdo, a tecla Alt esquerdo, e a tecla Delete pressionadas juntas a partir do console. No sistema padrão, isso fará com que o sistema reinicie. Voc precisa modificar o arquivo `/etc/inittab` para ter o comando `shutdown` com a opção `-h` como descrito em 'Instalar mais alguns pacotes após a instalação inicial' on page 43.

[8] Isso é porque o sistema Debian é, mesmo logo após a instalação padrão, configurado com as permissões de arquivos apropriadas que impedem que um usuário não privilegiado danifique o sistema. É claro, podem haver ainda alguns furos que podem ser explorados mas aqueles que se preocupam com esse problema não devem estar lendo essa seção mas devem estar lendo o Securing Debian Manual (`http://www.debian.org/doc/manuals/securing-debian-howto/`).

[9] A tecla Ctrl esquerdo e a tecla d pressionadas juntas. Não há necessidade de pressionar a tecla Shift mesmo quando esses caracteres de controle são referidos como "control D" com maiúscula.

[10] Se você entrar `root` ao invés de *`pinguim`* aqui e a senha correspondente, você obterá o acesso à conta `root`. Esse procedimento será necessário para obter novamente o acesso à conta `root`.

Use esse comando para explorar o sistema Debian. Essa é a melhor maneira de aprender. Por favor, explore alguns locais chave simplesmente usando as teclas de cursor e a tecla Enter:

- `/etc` e seus subdiretórios.
- `/var/log` e seus subdiretórios.
- `/usr/share/doc` e seus subdiretórios.
- `/sbin` e `/bin`

4.2.1 Melhorar o MC

Para fazer com que o MC mude o diretório de trabalho ao sair, você precisa modificar seu arquivo `~/.bashrc` (ou `/etc/bash.bashrc`, chamado a partir do `.bashrc`), como detalhado em sua página de manual, `mc(1)`, sob a opção `-P`. [11]

4.2.2 Iniciar o MC

```
$ mc
```

O MC cuida de todas as operações de arquivo através de seu menu, requerendo esforço mínimo do usuário. Simplesmente pressione a F1 para obter a tela de ajuda. Você pode brincar com o MC simplesmente pressionando as teclas de cursor e as teclas de função. [12]

4.2.3 Gerenciador de arquivos no MC

O padrão são dois painéis de diretórios contendo listas de arquivos. Outro modo útil é definir a janela direita para “info” para ver a informação de privilégio de acesso dos arquivos, etc. A seguir estão alguns comandos de teclado essenciais. Com o daemon `gpm` sendo executado, pode-se usar o mouse também. (Não esqueça de pressionar a tecla shift para ter o comportamento normal de cortar e colar no MC.)

- F1: Menu de ajuda
- F3: Visualizador de arquivos interno
- F4: Editor interno
- F9: Ativar o menu
- F10: Sair do Midnight Commander
- Tab: Mover entre as 2 janelas
- Insert: Marcar o arquivo para operação de múltiplos arquivos como cópia
- Del: Apagar o arquivo (Tenha cuidado — configure o MC para o modo de eliminação segura.)
- Teclas de cursor: Auto explicativo

---

[11] Se você não compreender exatamente o que estou dizendo aqui, você pode fazer isso mais tarde.

[12] Se estiver em um terminal, tal como `kon` e `kterm` para japonês, há alguns problemas com certos caracteres gráficos, e adicionar um `-a` à linha de comando do MC pode ajudar prevenir problemas.

4.2.4 Truques de linha de comando no MC

- Qualquer comando `cd` mudará o diretório exibido na janela selecionada.
- Ctrl-Enter ou Alt-Enter copiará um nome de arquivo para a linha de comando. Use isso com os comandos `cp` ou `mv` junto com edição de linha de comando.
- Alt-Tab exibirá possibilidades de expansão de um nome de arquivo no shell.
- Pode-se especificar o diretório inicial para ambas as janelas como argumentos para o MC; por exemplo, `mc /etc /root`.
- Esc + *tecla_numérica* == F*n* (isto é, Esc + '1' = F1, etc.; Esc + '0' = F10)
- Tecla Esc == tecla Alt (= Meta, M-); isto é, tecle Esc + 'c' se quiser Alt-c.

4.2.5 Editor no MC

O editor interno tem um interessante esquema de cortar-e-colar. Pressionando F3, marca-se o início da seleção, um segundo F3 marca o fim e destaca a seleção. Você pode então mover seu cursor. Se você pressionar F6, a área selecionada será movida para o local do cursor. Se pressionar F5, a área selecionada será copiada e inserida no local do cursor. Para salvar o arquivo, tecle F2. Para sair, use o F10. Muitas teclas de cursor funcionam intuitivamente.

Esse editor pode ser iniciado diretamente em um arquivo:

```
$ mc -e nome_do_arquivo_a_editar
$ mcedit nome_do_arquivo_a_editar
```

Esse editor não é multi janela, mas pode-se usar vários consoles Linux para obter o mesmo efeito. Para copiar entre janelas, use as teclas Alt-F*n* para alternar entre os consoles virtuais e use "Arquivo->Inserir arquivo" ou "Arquivo->copiar para Arquivo" para mover uma parte de um arquivo para outro.

Esse editor interno pode ser substituído por qualquer editor externo de sua escolha.

Além disso, muitos programas usam as variáveis de ambiente `EDITOR` ou `VISUAL` para decidir qual editor a ser usado. Se você não se sentir confortável com o `vim`, defina essas variáveis para o `mcedit` adicionando essas linhas ao arquivo `~/.bashrc`:

```
...
export EDITOR=mcedit
export VISUAL=mcedit
...
```

Eu recomendo que elas sejam definidas para `vim` se possível. Aprender os comandos do `vim` é a coisa certa a fazer, já que o editor Vi está sempre lá no mundo Linux/Unix. [13]

---

[13] Na verdade, o `vi` ou o `nvi` são os programas que você encontra em todo lugar. Ao invés disso eu escolhi o `vim` para o novato já que ele lhe oferece ajuda através da tecla F1 ao mesmo tempo que é similar o bastante e mais poderoso.

4.2.6 Visualizador no MC

Esse é um visualizador muito esperto. É uma grande ferramenta para procurar palavras em documentos. Eu o uso sempre para os arquivos no diretório `/usr/share/doc`. Essa é a maneira mais rápida de navegar entre as massas de informação do Linux. Esse visualizador pode ser iniciado diretamente assim:

```
$ mc -v nome_do_arquivo_a_visualizar
```

4.2.7 Recursos de auto início no MC

Pressione Enter em um arquivo, e o programa apropriado manipulará seu conteúdo. Esse é um recurso muito conveniente do MC.

```
arquivo executável:   Executa o comando
arquivo man, html:    Direciona o conteúdo para o programa visualizador
arquivo tar.gz, deb:  Navega em seu conteúdo com se fosse um subdiretório
```

Para permitir que esses recursos de visualizador e arquivo virtual funcionem, os arquivos visualizáveis não devem estar definidos como executáveis. Mude seus estados usando o comando `chmod` através do menu de arquivo do MC.

4.2.8 Sistema de arquivo virtual FTP no MC

O MC pode ser usado para acessar arquivos na Internet usando FTP. Vá para o menu pressionando F9, e então tecle 'p' para ativar o sistema de arquivos virtual FTP. Entre uma URL na forma `nomeusuario:senha@nomemaquina.nomedominio`, e será obtido um diretório remoto, que parecerá como um local.

Tente `http.us.debian.org/debian` como URL e navegue o repositório de arquivos do Debian. Veja 'Os repositórios Debian' on page 5 para detalhes sobre como eles são organizados.

4.3 Ambiente de trabalho semelhante ao Unix

Apesar de o MC possibilitar que você possa fazer quase tudo, é muito importante para você que aprenda como usar as ferramentas de linha de comando invocadas a partir do prompt shell e se torne familiar com o ambiente de trabalho semelhante ao Unix. [14]

---

[14] Nesse capítulo de tutorial, o shell significa `bash`. Para mais detalhes sobre os diferentes shells, veja 'Shell' on page 225.

4.3.1 Comandos de teclado especiais

No ambiente similar ao Unix, há alguns comandos de teclado que tem significados especiais. [15]

- Ctrl-U: Apaga uma linha antes do cursor.
- Ctrl-H: Apaga um caracter antes do cursor.
- Ctrl-D: Finaliza a entrada. (sai do shell se você estiver usando shell)
- Ctrl-C: Finaliza o programa sendo executado.
- Ctrl-Z: Pára temporariamente o programa. (coloca-o para trabalho de plano de fundo (background job), veja '`comando &`' on page 61)
- Ctrl-S: Trava (Halt) a saída para a tela. [16]
- Ctrl-Q: Reativa saída para a tela.

O shell padrão, `bash`, tem capacidades de edição de histórico e tab-completar para ajudar o uso interativo.

- seta-cima: Inicia busca no histórico de comandos.
- Ctrl-R: Inicia busca incremental no histórico de comandos.
- TAB: Completa a entrada do nome de arquivo para a linha de comando.
- Ctrl-V TAB: Entra TAB sem expansão para a linha de comando.

Outros comandos de teclado importantes a lembrar:

- Ctrl-Alt-Del: Reinicia/halt o sistema, veja 'Instalar mais alguns pacotes após a instalação inicial' on page 43.
- Click-botão-esquerdo-e-arrasta do mouse: Seleciona e copia para a área de tranferência (clipboard).
- Click botão do meio do mouse: Cola área de transferência no cursor.
- A tecla Meta (terminologia do Emacs) é atribuída tradicionalmente à tecla Alt esquerda. Alguns sistemas podem ser configurados para usar a tecla Windows como a tecla Meta.

Aqui, para usar um mouse no console de caracter do Linux, você precisa ter instalado o `gpm` como daemon. [17] Veja 'Configuração do mouse' on page 36.

4.3.2 Comandos Unix básicos

Vamos aprender os comandos Unix básicos. [18] Teste todos os comandos seguintes a partir da conta de usuário não privilegiado *`pinguim`*:

- `pwd`
  - Mostra o nome do diretório atual/de trabalho.
- `whoami`
  - Mostra o nome do usuário atual.
- `file` *`foo`*

---

[15] Em um console de caracter Linux normal, somente a tecla Ctrl e a tecla Alt esquerdas funcionam como esperado.

[16] Você pode desabilitar esse recurso de terminal usando `stty(1)`.

[17] No ambiente X Window, o mouse funciona da mesma forma com o programa Xterm.

[18] Aqui eu uso "Unix" em seu senso genérico. Quaisquer SOs clones do Unix geralmente oferecem os comandos equivalentes. O sistema Debian não é uma exceção. Não se preocupe se alguns comandos não funcionarem como você deseja agora. Esses exemplos não precisam ser executados nessa ordem.

  - Mostra o tipo de arquivo do arquivo *foo*.
- `type -p` *`nomedecomando`*
  - Mostra uma localização de arquivo do comando *`nomedecomando`*.
  - `which` *`nomedecomando`* faz o mesmo.
- `type` *`nomedecomando`*
  - Mostra informação sobre o comando *`nomedecomando`*.
- `apropos` *`palavra-chave`*
  - Encontra comandos relacionados a *`palavra-chave`*.
  - `man -k` *`palavra-chave`* faz o mesmo.
- `whatis` *`nomedecomando`*
  - Mostra uma explicação de uma linha sobre o comando *`nomedecomando`*.
- `man -a` *`nomedecomando`*
  - Mostra uma explicação curta sobre o comando *`nomedecomando`*. (estilo Unix)
- `info` *`nomedecomando`*
  - Mostra uma explicação longa sobre o comando *`nomedecomando`*. (estilo GNU)
- `ls`
  - Lista o conteúdo do diretório. (arquivos não-ponto e diretórios) [19]
- `ls -a`
  - Lista o conteúdo do diretório. (toods os arquivos e diretórios)
- `ls -A`
  - Lista o conteúdo do diretório. (quase todos os arquivos e diretórios, isto é, pula "`..`" e "`.`")
- `ls -la`
  - Lista todo o conteúdo do diretório com informação de detalhe. Veja 'O conceito de sistema de arquivos no Debian' on page 67.
- `ls -d`
  - Lista todos os diretórios sob o diretório atual.
- `lsof` *`foo`*
  - Lista o estado de abertura do arquivo *`foo`*.
- `mkdir` *`foo`*
  - Cria um novo diretório *`foo`* no diretório atual.
- `rmdir` *`foo`*
  - Remove um diretório *`foo`* no diretório atual.
- `cd` *`foo`*
  - Muda para o diretório *`foo`* no diretório atual ou no diretório listado na variável `CDPATH`. Veja o comando `cd` na página de manual `builtins(7)`
- `cd /`
  - Muda o diretório para o diretório raiz
- `cd`
  - Muda o diretório para o diretório home do usuário.
- `cd /`*`foo`*
  - Muda o diretório para o caminho de diretório absoluto `/`*`foo`*.
- `cd ..`

---

[19] O Unix tem uma tradição de ocultar nomes de arquivos que começam com "`.`". Eles são tradicionalmente arquivos que contêm informação de configuração e preferências do usuário.

  - Muda o diretório para o diretório pai.
- `cd ~`*`foo`*
  - Muda o diretório para o diretório home do usuário *`foo`*.
- `cd -`
  - Muda o diretório para o diretório anterior.
- `</etc/motd pager`
  - Mostra o conteúdo de `/etc/motd` usando o paginador padrão. Veja '`comando <` *`foo`*' on page 62. [20]
- `touch` *`arquivolixo`*
  - Cria um arquivo *`arquivolixo`* vazio.
- `cp` *`foo bar`*
  - Copia um arquivo *`foo`* existente para um novo arquivo *`bar`*.
- `rm` *`arquivolixo`*
  - Remove um arquivo *`arquivolixo`*.
- `mv` *`foo bar`*
  - Renomeia um arquivo *`foo`* existente para um novo nome *`bar`*.
- `mv` *`foo bar/baz`*
  - Move um arquivo *`foo`* existente para um novo local com um novo nome *`bar/baz`*. O diretório *`bar/`* precisa existir.
- `chmod 600` *`foo`*
  - Torna um arquivo *`foo`* existente não-legível e não-gravável para outras pessoas. (não-executável para todos)
- `chmod 644` *`foo`*
  - Torna um arquivo *`foo`* existente legível mas não-gravável por outras pessoas. (não-executável para todos)
- `chmod 755` *`foo`*
  - Torna um arquivo *`foo`* existente legível mas não-gravável por outras pessoas. (executável para todos)
- `top`
  - Mostra a informação de processos usando tela cheia. Pressione "q" para sair
- `ps aux | pager`
  - Mostra informação sobre todos os processos sendo executados usando saída de estilo BSD. Veja '`comando1 | comando2`' on page 61.
- `ps -ef | pager`
  - Mostra informação sobre todos os processos sendo executados usando saída estilo Unix system-V.
- `ps aux | grep -e "[e]xim"`
  - Mostra todos os processos executando `exim`. Aprenda a expressão regular a partir da página de manual do `grep(1)` digitando `man grep`.
- `ps axf | pager`
  - Mostra informação sobre todos os processos sendo executados com saída com ASCII art.
- `kill` *`1234`*

---

[20] O paginador padrão do sistema Debian pelado é o `more` que não pode rolar para trás. Instalando o pacote `less` usando a linha de comando `apt-get install less`, o `less` se torna o paginador padrão e você pode rolar para trás usando as teclas de cursor.

  - Mata um processo identificado pela ID de processo: *1234*. Veja 'Matar um processo' on page 117.
- `grep -e "padrão" *.html`
  - Encontra um *"padrão"* em todos os arquivos terminando com `.html` no diretório atual e mostra todos.
- `gzip foo`
  - Comprime `foo` para criar `foo.gz` usando o código Lempel-Ziv (LZ77).
- `gunzip foo.gz`
  - Descompacta `foo.gz` para criar o arquivo `foo`.
- `bzip2 foo`
  - Compacta `foo` para criar `foo.bz2` usando o algoritmo de compressão de texto por ordenação de bloco de Burrows-Wheeler, e código de Huffman. (Melhor compressão que o `gzip`)
- `bunzip2 foo.bz2`
  - Descompacta `foo.bz2` para criar o arquivo `foo`.
- `tar -xvvf foo.tar`
  - Extrai arquivos a partir do arquivo `foo.tar`.
- `tar -xvvzf foo.tar.gz`
  - Extrai arquivos a partir do arquivo `foo.tar.gz` gzipado.
- `tar -xvvf --bzip2 foo.tar.bz2`
  - Extrai arquivos a partir do arquivo `foo.tar.bz2`.[21]
- `tar -cvvf foo.tar bar/`
  - Arquiva o conteúdo da pasta `bar/` no arquivo `foo.tar`.
- `tar -cvvzf foo.tar.gz bar/`
  - Arquiva o conteúdo da pasta `bar/` no arquivo comprimido `foo.tar.gz`.
- `tar -cvvf --bzip2 foo.tar.bz2 bar/`
  - Arquiva o conteúdo da pasta `bar/` no arquivo `foo.tar.bz2`.[22]
- `zcat README.gz | pager`
  - Mostra o conteúdo do `README.gz` comprimido usando o paginador padrão.
- `zcat README.gz > foo`
  - Cria um arquivo `foo` com o conteúdo descompactado de `README.gz`.
- `zcat README.gz >> foo`
  - Acrescenta o conteúdo descomprimido de `README.gz` ao final do arquivo `foo`. (Se ele não existir, cria-o primeiro.)
- `find . -name padrão`
  - encontra nomes de arquivo que combinem usando o `padrão` shell. (mais lento)
- `locate -d . padrão`
  - encontra nomes de arquivo que combinem usando o `padrão` shell. (mais rápido usando uma base de dados gerada regularmente)

Por favor cruze diretórios e mergulhe no sistema usando os comandos acima como um treinamento. Se você tiver questões sobre qualquer um dos comandos de console, por favor, certifique-se de ler a página de manual. Por exemplo, esses comandos são um bom começo:

---

[21] `--bzip2` é usado aqui ao invés da nova opção curta `-j` para garantir que isso funcione com a versão antiga do `tar` no Potato.

[22] `--bzip2` é usado aqui novamente para garantir compatibilidade.

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

Também é uma boa hora para iniciar o `vim` e pressionar a tecla F1. Você deve ler pelo menos as primeiras 35 linhas. Então faça o curso de treinamento online movendo o cursor para |`tutor`| e pressionando Ctrl-]. Veja 'Editores' on page 211 para aprender mais sobre editores.

Por favor note que muitos comandos similares a Unix incluindo os do GNU e BSD mostrarão uma breve informação de ajuda se você invocá-los em uma das seguintes maneiras (ou sem nenhum argumento em alguns casos):

```
$ nomedecomando --help
$ nomedecomando -h
```

Teste também exemplos em 'Dicas Debian' on page 109 como seu auto treinamento.

4.3.3 A execução de comando

Agora você tem alguma idéia de como usar o sistema Debian. Vamos olhar com mais profundidade o mecanismo da execução de comando no sistema Debian. [23]

4.3.4 Comando simples

Um comando simples é uma seqüência de

1. atribuições de variáveis (opcional)
2. nome de comando
3. argumentos (opcional)
4. redireções (opcional:> , >> , < , << , etc.)
5. operador de controle (opcional: `&&` , || ; **<newline>** , `;` , `&` , `(` , `)`)

Para comandos mais complexos com aspas e substituições, veja 'Processamento de linha de comando' on page 229.

4.3.5 Execução de comando e variável de ambiente

A execução de comando típica usa uma seqüência de linha shell como a seguinte: [24]

---

[23] Aqui, eu simplifiquei a realidade para o novato. Veja a página de manual `bash(1)` para ter uma explicação exata.

[24] Para obter a saída seguinte, você precisa instalar o locale francês, veja 'Locales' on page 172. Isso não é essencial para o tutorial. Isso é feito apenas para indicar seus efeitos potenciais.

```
$ date
Sun Oct 26 08:17:20 CET 2003
$ LC_ALL=fr_FR date
dim oct 26 08:17:39 CET 2003
```

Aqui, o programa `date` é executado como o trabalho em primeiro plano (foreground). A variável de ambiente `LC_ALL` é:

- indefinida (padrão do sistema, o mesmo que `C` para o primeiro comando
- definida para `fr_FR` (locale francês) para o segundo comando

Muitas execuções de comandos geralmente não têm uma definição de variável de ambiente precedendo. Para o exemplo acima, você pode executar alternativamente:

```
$ LC_ALL=fr_FR
$ date
dim oct 26 08:17:39 CET 2003
```

Como você pode ver aqui, a saída do comando é afetada pela variável de ambiente para produzir saída em francês. Se você quiser que a variável de ambiente seja transmitida aos subprocessos (por exemplo, quando chamando um script shell), você precisa ao invés "export"á-la usando:

```
$ export LC_ALL
```

4.3.6 Caminho de busca de comando

Quando você digita um comando no shell, o shell procura o comando na lista de diretórios contida pela variável de ambiente `PATH`. O valor da variável de ambiente `PATH` também é chamado de caminho de busca do shell.

Na instalação Debian padrão, a variável de ambiente `PATH` de uma conta de usuário não deve incluir `/sbin`. Então se você quiser executar quaisquer comandos como `ifconfig` de `/sbin`, você precisa mudar a variável de ambiente `PATH` para incluí-lo. A variável de ambiente `PATH` geralmente é definida pelo arquivo de inicialização `~/.bash_profile`, veja 'Configuração do bash' on page 36.

4.3.7 Opções de linha de comando

Alguns comandos levam argumentos. Os argumentos começando com `-` ou `--` são chamados de opções e controlam o comportamento do comando.

```
$ date
Mon Oct 27 23:02:09 CET 2003
$ date -R
Mon, 27 Oct 2003 23:02:40 +0100
```

Aqui o argumento de linha de comando `-R` muda o comportamento do comando `date` para saída com data em conformidade com a RFC-2822.

4.3.8 Curingas (wildcards) de shell

Freqüentemente se deseja que um comando funcione com um grupo de arquivos sem digitar todos eles. O padrão de expansão de nome de arquivo usando os **curingas** do shell facilitam essas necessidades.

- `*`
  - Substitui qualquer grupo de 0 ou mais caracteres.
  - Não substitui um nome de arquivo que comece com “`.`”.
- `?`
  - Substitui exatamente um caracter.
- `[...]`
  - Substitui exatamente um caracter por qualquer caracter colocado entre os colchetes.
- `[a-z]`
  - Substitui exatamente um caracter por qualquer caracter entre `a` e `z`.
- `[^...]`
  - Substitui exatamente um caracter por qualquer outro diferente dos caracteres entre colchetes (excluindo “`^`”).

Por exemplo, teste o seguinte e pense por si mesmo:

```
$ mkdir lixo; cd lixo; $ touch 1.txt 2.txt 3.c 4.h .5.txt
$ echo *.txt
1.txt 2.txt
$ echo *
1.txt 2.txt 3.c 4.h
$ echo *.[hc]
3.c 4.h
$ echo .*
. .. .5.txt
$ echo .[^.]*
.5.txt
$ echo [^1-3]*
4.h
$ cd ..; rm -rf lixo
```

4.3.9 Valor de retorno do comando

Cada comando retorna seu estado de saída como o valor de retorno.

- valor de retorno = 0 se o comando é executado com sucesso.
- valor de retorno = não-zero se o comando finaliza com erro.

O valor de retorno pode ser acessado pela variável shell `$?` imediatamente após a execução.

Por favor note que, quando o valor de retorno é usado no contexto lógico para o shell, **sucesso** é tratado como o **TRUE** (VERDADEIRO) lógico. Isso é um pouco não intuitivo já que **sucesso** produz valor **zero**.

Veja 'Condicionais de shell' on page 228.

4.3.10 Seqüências de comando típicas

Vamos tentar lembrar as seguintes expressões de comandos shell. Veja 'Parâmetros shell' on page 227, 'Redirecionamento shell' on page 227, 'Condicionais de shell' on page 228, e 'Processamento de linha de comando' on page 229 depois de ler essas expressões.

`comando &`

O `comando` é executado na subshell no **plano de fundo** (background). Tarefas em plano de fundo permitem que usuários executem vários programas em um único shell.

O gerenciamento de processos em plano de fundo envolve os comandos embutidos do shell: `jobs`, `fg`, `bg`, e `kill`. Por favor leia as seções da página de manual `bash(1)` em "SIGNALS", "JOB CONTROL", e "SHELL BUILTIN COMMANDS". [25]

`comando1 | comando2`

A saída padrão do `comando1` é transmitida à entrada padrão do `comando2`. Ambos os comandos podem estar sendo executados **ao mesmo tempo**. Isso é chamado de **pipeline**.

`comando1 ; comando2`

O `comando1` e o `comando2` são executados **seqüencialmente**.

`comando1 && comando2`

O `comando1` é executado. Se tiver sucesso, o `comando2` também é executado **seqüencialmente**. Retorna sucesso se o `comando1` **ou** o `comando2` tiverem sucesso.

`comando >` *`foo`*

Redireciona a saída padrão do `comando` para um arquivo *`foo`*. (sobrescreve)

---

[25] O sistema Debian é um sistema multi-tarefa.

`comando >> foo`

Redireciona a saída padrão do `comando` para um arquivo *`foo`* (acrescenta (append))

`comando > foo 2>&1`

Redireciona tanto a saída padrão quanto a saída padrão de erro do `comando` para um arquivo *`foo`*.

`comando < foo`

Redireciona a entrada padrão do `comando` para um arquivo *`foo`*. Teste:

```
$ </etc/motd pager
 ... (as saudações)
$ pager </etc/motd
 ... (as saudações)
$ pager /etc/motd
 ... (as saudações)
$ cat /etc/motd | pager
 ... (as saudações)
```

Apesar de as 4 sintaxes mostrarem a mesma coisa, o último exemplo executa um comando `cat` extra e desperdiça recursos sem motivo.

4.3.11 Apelido (alias) de comando

Você pode definir um apelido (alias) para o comando usado freqüentemente. Por exemplo:

```
$ alias la='ls -la'
```

Agora, `la` funciona como um atalho para `ls -la` que lista todos os arquivos em formato de listagem longa.

Você pode identificar o caminho (path) exato ou identidade do comando usando o comando `type`. Por exemplo:

```
$ type ls
ls is hashed (/bin/ls)
$ type la
la is aliased to `ls -la'
$ type echo
echo is a shell builtin
$ type file
file is /usr/bin/file
```

Aqui o `ls` foi procurado recentemente enquanto `file` não foi, então o `ls` está "hashed", isto é, o shell tem um registro interno para o acesso rápido da localização do comando `ls`.

4.4 Processamento de texto em similares a Unix

Há algumas ferramentas de processamento de texto que são usadas muito freqüentemente no sistema similar ao Unix.

- Nenhuma expressão regular é usada:
  - `head` tem como saída a primeira parte de arquivos.
  - `tail` tem como saída a última parte de arquivos.
  - `sort` ordena linhas de arquivos texto.
  - `uniq` remove linhas duplicadas de um arquivo ordenado.
  - `tr` troca ou apaga caracteres.
  - `diff` compara aquivos linha por linha.
- É usada Expressão regular básica (ERB):
  - `grep` encontra texto com o padrão.
  - `ed` é um editor de linha primitivo.
  - `sed` é um editor de fluxo.
  - `vi` é um editor de tela.
  - `emacs` é um editor de tela.
- É usada Expressão regular extendida (ERE):
  - `egrep` encontra texto com o padrão.
  - `awk` faz processamento de texto simples. Veja 'Awk' on page 230.
  - `perl` faz todo tipo de processamento de texto concebível. Veja 'Perl' on page 231.

Veja 'Substituição de expressões regulares' on page 122, 'Pequenos scripts para ligar comandos' on page 124, e 'Loucura de script Perl curto' on page 126 para ter alguns exemplos de script.

4.4.1 Expressões regulares

Expressões regulares são usadas em muitas ferramentas de processamento de texto. Elas são análogas aos curingas de shell (veja 'Curingas (wildcards) de shell' on page 60), mas são também mais complicadas e mais poderosas.

A expressão regular descreve um padrão de combinação e é feita de caracteres de texto e **metacaracteres**. O metacaracter é apenas um caracter com um significado especial. Há 2 estilos principais, ERB e ERE, dependendo das ferramentas de texto como descrito em 'Processamento de texto em similares a Unix' on the current page.

Para as EREs, os **metacaracteres** incluem "`\ . [ ] ^ $ * + ? ( ) { } |`". A expressão regular significa:

- `c`
  - Combina com o não-metacaracter "`c`".
- `\c`
  - Combina com o caracter literal "`c`".
- `.`

  - Combina com qualquer caracter incluindo newline (nova linha).
- `^`
  - Combina com o começo de uma string (cadeia de caracteres).
- `$`
  - Combina com o final de uma string.
- `\<`
  - Combina com o começo de uma palavra.
- `\>`
  - Combina com o final de uma palavra.
- `[abc...]`
  - Essa lista de caracteres combina com qualquer um dos caracteres “`abc...`”.
- `[^abc...]`
  - Essa lista de caracteres negada combina com qualquer caracter exceto “`abc...`”.
- `r*`
  - Combina com zero ou mais expressões regulares identificadas por “`r`”.
- `r+`
  - Combina com uma ou mais expressões regulares identificadas por “`r`”.
- `r?`
  - Combina com zero ou uma expressões regulares identificadas por “`r`”.
- `r1|r2`
  - Combina com uma das expressões regulares identificadas por “`r1`” ou “`r2`”.
- `(r1|r2)`
  - Combina com uma das expressões regulares identificadas por “`r1`” ou “`r2`” e a trata como uma expressão regular **entre parênteses**.

Em ERBs os **metacaracteres** “`+ ?  ( ) { } |`” perdem seu significado especial; ao invés disso use as versões com barra invertida “`\+ \?  \( \) \{ \} \|`”. Assim a construção de agrupamento `(r1|r2)` precisa ser colocada como `\(r1|r2\)` em ERBs. No `emacs`, apesar de ser basicamente ERB, “`+ ?`” são tratados como os **metacaracteres**. Então não há necessidade de usar barra invertida para eles. Veja ‘Expressões de substituição’ on this page para saber como a construção de agrupamento é usada.

Por exemplo, o `grep` pode ser usado para fazedr a busca de texto usando a expressão regular:

```
$ egrep 'GNU.*LICENSE|Yoyodyne' /usr/share/common-licenses/GPL
                    GNU GENERAL PUBLIC LICENSE
                    GNU GENERAL PUBLIC LICENSE
  Yoyodyne, Inc., hereby disclaims all copyright interest in the program
```

4.4.2 Expressões de substituição

Para a expressão de substituição, os seguintes caracteres têm significados especiais:

- `&`
  - Representa o que a expressão regular casou (matched). (use `\&` no `emacs`)

- `\n`
  - Representa o que a *n*-ésima expressão regular **entre parênteses** casou.

Para string de substituição Perl, usa-se `$n` ao invés de `\n` e `&` não tem significado especial.

Por exemplo:

```
$ echo zzz1abc2efg3hij4 | \
  sed -e ’s/\(1[a-z]*\)[0-9]*\(.*\)$/=&=/’
zzz=1abc2efg3hij4=
$ echo zzz1abc2efg3hij4 | \
  sed -e ’s/\(1[a-z]*\)[0-9]*\(.*\)$/\2===\1/’
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
  perl -pe ’s/(1[a-z]*)[0-9]*(.*)$/$2===$1/’
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
  perl -pe ’s/(1[a-z]*)[0-9]*(.*)$/=&=/’
zzz=&=
```

Aqui por favor preste uma atenção extra ao estilo da expressão regular **entre parênteses** e como as strings casadas são usadas no processo de substituição de texto em ferramentas diferentes.

Essas expressões regulares podem ser usadas para os movimentos de cursor e as ações de substituições de texto nos editores também.

Por favor leia todas as páginas de manual relacionadas para aprender esses comandos.

4.5 Sistema de arquivos em similares ao Unix

No GNU/Linux e outros SOs similares ao Unix, os **arquivos** são organizados em **diretórios**. [26] Todos os **arquivos** e **diretórios** são arranjados em uma grande árvore, a hierarquia de arquivos, com raiz em `/`.

Esses arquivos e diretórios podem ser distribuídos entre vários dispositivos. O comando `mount(8)` serve para anexar o sistema de arquivos encontrado em um dispositivo à grande árvore de arquivos. Inversamente, o comando `unmount(8)` desconecta o dispositivo novamente.

4.5.1 Básico de arquivos Unix

Aqui está o básico:

---

[26] **Diretórios** são chamados de **pastas** em alguns outros sistemas.

- Nomes de arquivos são sensíveis a maiúsculas/minúsculas. Isto é, `MEUARQUIVO` e `MeuArquivo` são arquivos **diferentes**.
- O diretório raiz (root) é referenciado simplesmente como `/`. Não confunda essa "raiz"(root) com o usuário root. Veja 'Login em um prompt shell como root' on page 47.
- Todo diretório tem um nome que pode conter quaisquer letras ou símbolos **exceto** `/`. [27] O diretório raiz é uma exceção; seu nome é `/` (pronunciado como "slash" (barra) ou "o diretório raiz") e não pode ser renomeado.
- Cada arquivo ou diretório é designado por um **nome de arquivo totalmente qualificado**, **nome de arquivo absoluto**, ou um **caminho**, dando a seqüência de diretórios através dos quais se precisa passar para alcançá-lo. Os três termos são sinônimos. Todos os nomes de arquivo absolutos começam com o diretório `/`, e há uma `/` entre cada diretório ou arquivo no nome de arquivo. A primeira `/` é o nome de um diretório, mas as outras são simplesmente separadores para distinguir as partes do nome de arquivo. As palavras usadas aqui podem ser confusas. Tome o seguinte exemplo:

  ```
  /usr/share/keytables/us.map.gz
  ```

  Esse é um nome de arquivo totalmente qualificado; algumas pessoas o chamam de **caminho** (path). Entretanto, as pessoas também se referem a `us.map.gz` sozinho como um nome de arquivo. [28]
- O diretório raiz tem um número de ramos, como `/etc` e `/usr`. Esses subdiretórios também ramificam em ainda mais subdiretórios, como `/etc/init.d` e `/usr/local`. O conjunto todo junto é chamado de **árvore de diretórios**. Você pode pensar em um nome de arquivo absoluto como uma rota a partir da base da árvore (`/`) até o fim de algum ramo (um arquivo). Você também ouvirá as pessoas falarem sobre a árvore de diretórios como se fosse uma árvore **familiar**: assim os subdiretórios tem **ancestrais**, e um caminho mostra a ascendência completa de um arquivo. Há também caminhos relativos que começam em algum lugar diferente do diretório raiz. Você deve se lembrar de que o diretório `../` se refere ao diretório pai.
- Não há diretório que corresponda a um dispositivo físico, tal como seu disco rígido. Isso difere do CP/M, DOS, e Windows, onde todos os caminhos começam com um nome de dispositivo como `C:\`. Veja 'O conceito de sistema de arquivos no Debian' on the next page.

As melhores práticas detalhadas para a hierarquia de arquivos estão descritas no Padrão de Hierarquia de Sistema de Arquivos (`/usr/share/doc/debian-policy/fhs/fhs.txt.gz`). Você deve se lembrar dos seguintes fatos como iniciante:

- `/`
  - Uma `/` simples representa o diretório raiz.
- `/etc`
  - Esse é o lugar para os arquivos de configuração globais do sistema.
- `/var/log`

---

[27] Apesar de você **poder** usar praticamente quaisquer letras ou símbolos em um nome de arquivo, na prática isso é uma má idéia. É melhor evitar quaisquer caracteres que têm significados especiais freqüentemente na linha de comando, incluindo espaços, tabs, newlines e outros caracteres especiais: `{ } ( ) [ ] ' ` " \ / > < | ; ! # & ^ * % @ $`. Se você quiser separar palavras em um nome, o ponto, hífen, e subscrito (underscore) são boas escolhas. Você também pode colocar maiúsculas nas letras iniciais de cada palavra, `DesseJeito`.

[28] Há também outro uso para a palavra **caminho**. Veja 'Caminho de busca de comando' on page 59. O significado desejado geralmente fica claro pelo contexto.

  - Esse é o lugar dos arquivos de log do sistema.
- `/home`
  - Esse é o diretório que contém todos os diretórios home para todos os usuários não privilegiados.

4.5.2 O conceito de sistema de arquivos no Debian

Seguindo a tradição Unix, o sistema Debian fornece o sistema de arquivos sob o qual os dados físicos em discos rígidos e outros dispositivos de armazenamento, e a interação com os dispositivos de hardware como telas de console e consoles seriais remotos são representados em uma forma unificada.

Cada arquivo, diretório, pipe nomeada, ou dispositivo físico em um sistema Debian tem uma estrutura de dado chamada **inode** que descreve seus atributos associados como o usuário que é seu dono, o grupo ao qual pertence, a hora do último acesso, etc.. Veja `/usr/include/linux/fs.h` para ter a definição exata da `estrutura inode` no sistema Debian GNU/Linux.

Essa representação unificada de entidades físicas é muito poderosa já que nos permite usar o mesmo comando para o mesmo tipo de operação em muitos dispositivos totalmente diferentes.

Todos os seus arquivos poderiam estar em um disco — ou você poderia ter 20 discos, alguns deles conectados a computadores diferentes em algum lugar na rede. Você não pode dizer simplesmente olhando pela árvore de diretórios, e praticamente todos os comandos funcionam da mesma forma independentemente do tipo de dispositivo(s) físico(s) em que seus arquivos realmente estão.

4.5.3 Permissões de acesso de arquivos e diretórios

As permissões de acesso a arquivo e diretório são definidas separadamente para as três seguintes categorias de usuários afetados:
- o **usuário** (user) que é dono do arquivo (u),
- outros usuários no **grupo** (group) ao qual o arquivo pertence (g), e
- todos os **outros** (other) usuários (o).

Para um arquivo, cada permissão correspondente possibilita:
- **ler** (r) (read): para examinar o conteúdo do arquivo,
- **escrever** (w) (write): para modificar o arquivo, e
- **executar** (x): para executar o arquivo como um comando.

Para um diretório, cada permissão correspondente possibilita:
- **ler** (r): para listar o conteúdo do diretório,
- **escrever** (w): para adicionar ou remover arquivos no diretório, e
- **executar** (x): para acessar arquivos no diretório.

Aqui, permissão de **executar** no diretório não significa apenas permitir a leitura dos arquivos em seu diretório, mas também permitir a visualização de seus atributos, como o tamanho e o tempo de modificação.

Para obter as informações de permissão (e outras) de arquivos e diretórios, usa-se o `ls`. Consulte `ls(1)`. Quando o `ls` é invocado com a opção `-l`, exibe essas informações na seguinte ordem:

- o **tipo de arquivo** (primeiro caracter)
  - `-`: arquivo normal
  - `d`: diretório
  - `l`: ligação simbólica (symlink)
  - `c`: nó de dispositivo caracter
  - `b`: nó de dispositivo bloco
  - `p`: pipe nomeada
  - `s`: soquete
- as **permissões** de acesso do arquivo (os próximos nove caracteres, consistindo de três caracteres cada para usuário (user), grupo (group), e outros (other) nessa ordem)
- o **número de ligações diretas** (hard links) para o arquivo
- o nome do **usuário** que é dono do arquivo
- o nome do **grupo** ao qual o arquivo pertence
- o **tamanho** do arquivo em caracteres (bytes)
- a **data e hora** do arquivo (mtime)
- o **nome** do arquivo.

Para mudar o dono do arquivo, usa-se o `chown` a partir da conta root. Para mudar o grupo do arquivo, usa-se o `chgrp` a partir da conta do dono do arquivo ou da conta root. Para mudar as permissões de acesso de arquivo e diretório, usa-se o `chmod` a partir da conta do dono do arquivo ou da conta root. A sintaxe básica para manipular o arquivo `foo` é:

```
# chown novodono foo
# chgrp novogrupo foo
# chmod  [ugoa][+-=][rwx][,...] foo
```

Veja `chown(1)`,`chgrp(1)`,e `chmod(1)` para os detalhes.

Por exemplo, para fazer uma árvore de diretórios ser de propriedade de um usuário *foo* e compartilhada por um grupo *bar*, execute os seguintes comandos a partir da conta root:

```
# cd /algum/local/
# chown -R foo:bar .
# chmod -R ug+rwX,o=rX .
```

Há mais três bits de permissão especiais:

- **set user ID** (s ou S ao invés do x do usuário),
- **set group ID** (s ou S ao invés do x do grupo), e
- **sticky bit** (t ou T ao invés do x de outros).

Aqui a saída de `ls -l` para esses bits é transformada em maiúsculas se os bits de execução estiverem ocultos devido a essas saídas estarem desabilitadas.

Habilitar **set user ID** para um arquivo executável permite que um usuário execute o arquivo executável com ID do dono do arquivo (por exemplo **root**). Similarmente, definir **set group**

ID em um arquivo executável permite que um usuário execute o arquivo executável com o group ID do arquivo (por exemplo **root**). Devido a esses ajustes poderem causar sérios riscos de segurança, suas habilitações requerem cuidado extra.

Habilitar **set group ID** para um diretório permite o esquema de criação de arquivo no estilo BSD, onde todos os arquivos criados no diretório pertencem ao **grupo** do diretório.

Habilitar **sticky bit** em um diretório impede que um arquivo no diretório seja removido por um usuário que não é dono do arquivo. Para assegurar o conteúdo de um arquivo em diretórios globalmente graváveis como `/tmp` ou em diretórios graváveis por grupo, é necessário não apenas que a permissão de **escrita** (write) esteja desativada para o arquivo, mas também ativar o **sticky bit** no diretório. Caso contrário, o arquivo pode ser removido e um novo arquivo pode ser criado com o mesmo nome por qualquer usuário que tenha permissão de escrita no diretório.

Aqui estão alguns exemplos interessantes das permissões de arquivo.

```
$ ls -l /etc/passwd /etc/shadow /dev/ppp /usr/sbin/pppd
crw-rw----    1 root     dip      108,   0 Jan 18 13:32 /dev/ppp
-rw-r--r--    1 root     root         1051 Jan 26 08:29 /etc/passwd
-rw-r-----    1 root     shadow        746 Jan 26 08:29 /etc/shadow
-rwsr-xr--    1 root     dip        234504 Nov 24 03:58 /usr/sbin/pppd
$ ls -ld /tmp /var/tmp /usr/local /var/mail /usr/src
drwxrwxrwt    4 root     root         4096 Feb  9 16:35 /tmp
drwxrwsr-x   10 root     staff        4096 Jan 18 13:31 /usr/local
drwxrwsr-x    3 root     src          4096 Jan 19 08:36 /usr/src
drwxrwsr-x    2 root     mail         4096 Feb  2 22:19 /var/mail
drwxrwxrwt    3 root     root         4096 Jan 25 02:48 /var/tmp
```

Há um modo numérico alternativo para descrever permissões de arquivo em comandos `chmod(1)`. Esse modo numérico usa de 3 a 4 dígitos de números octais (base=8). Cada dígito corresponde a:

- primeiro dígito opcional: soma de **set user ID** (=4), **set group ID** (=2), e **sticky bit** (=1)
- segundo dígito: soma de permissões de **leitura** (=4), **escrita** (=2), e **execução** (=1) para o **usuário**
- terceiro dígito: idem para o **grupo**
- quarto dígito: idem para **outros**

Isso parece complicado mas na verdade é muito simples. Se você olhar para as primeiras (2-10) colunas da saída do comando `ls -l` e lê-las como uma representação binária (base=2) de permissões de arquivo ("-" sendo "0" e "rwx" sendo "1"), esse valor de modo numérico deve fazer sentido como uma representação octal (base=8) das permissões de arquivo para você. [29] Por exemplo, teste:

```
$ touch foo bar
```

---

[29] É claro que esse método funciona apenas para o modo numérico de 3 dígitos.

```
$ chmod u=rw,go=r foo
$ chmod 644 bar
$ ls -l foo bar
-rw-r--r--    1 pinguim  pinguim  0 Nov  3 23:30  foo
-rw-r--r--    1 pinguim  pinguim  0 Nov  3 23:30  bar
```

A máscara de permissão de arquivo padrão pode ser definida usando o comando interno de shell `umask`. Consulte `builtins(7)`.

4.5.4 Marcas de tempo

Há três tipos de marcas de tempo para um arquivo GNU/Linux:

- **mtime**: a hora da modificação (`ls -l`),
- **ctime**: a hora da mudança de estado (`ls -lc`), e
- **atime**: a hora do último acesso (`ls -lu`).

Note que **ctime** não é a hora de criação do arquivo.

- Sobrescrever um arquivo altera **mtime**, **ctime**, e **atime** do arquivo.
- Mudar a permissão ou o dono de um arquivo altera o **ctime** e o **atime** do arquivo.
- Ler um arquivo altera o **atime** do arquivo.

Note que até uma simples leitura de um arquivo no sistema Debian normalmente causará uma operação de escrita de arquivo para atualizar a informação de **atime** no **inode**. Montar um sistema de arquivos com a opção `noatime` fará com que o sistema pule essa operação e resultará em acesso para leitura mais rápido. Veja `mount(8)`.

Use o comando `touch(1)` para mudar as marcas de tempo de arquivos existentes.

4.5.5 Ligações (links)

Há dois métodos para associar um arquivo *foo* a um nome de arquivo diferente *bar*.

- uma **ligação direta** (hard link) é um nome duplicado para um arquivo existente. (`ln foo bar`),
- **ligação simbólica**, ou "symlink", é um arquivo especial que aponta para outro arquivo pelo nome (`ln -s foo bar`).

Veja no seguinte exemplo as mudanças no contador de ligações e as sutis diferenças no resultado do comando `rm`.

```
$ echo "Conteúdo Original" > foo
$ ls -l foo
-rw-r--r--    1 osamu    osamu          4 Feb  9 22:26 foo
$ ln foo bar     # ligação direta (hard link)
$ ln -s foo baz  # ligação simbólica (symlink)
$ ls -l foo bar baz
-rw-r--r--    2 osamu    osamu          4 Feb  9 22:26 bar
lrwxrwxrwx    1 osamu    osamu          3 Feb  9 22:28 baz -> foo
```

```
-rw-r--r--    2 osamu    osamu           4 Feb  9 22:26 foo
$ rm foo
$ echo "Novo Conteúdo" > foo
$ cat bar
Conteúdo Original
$ cat baz
Novo Conteúdo
```

A ligação simbólica tem as permissões nominais de acesso de arquivo como "rwxrwxrwx", como mostrado no exemplo acima, com as permissões de acesso efetivas ditadas pelas permissões do arquivo apontado por elas.

O diretório `.` liga ao diretório em que aparece, então o contador de ligações de qualquer diretório novo começa com 2. O diretório `..` liga ao diretório superior, então o contador de ligações do diretório aumenta com a adição de novos subdiretórios.

4.5.6 Pipes nomeados (FIFOs)

Um pipe nomeado é um arquivo que age como um pipe. Você coloca alguma coisa no arquivo e isso sai do outro lado. Então é chamado de FIFO, ou First-In-First-Out (Primeiro-Entra-Primeiro-Sai): a primeira coisa que você coloca no pipe é a primeira coisa que sai do outro lado.

Se você escrever em um pipe nomeado, o processo que está escrevendo no pipe não termina até que a informação que está sendo escrita seja lida pelo pipe. Se você ler de um pipe nomeado, o processo de leitura espera até que haja algo para ler antes de terminar. O tamanho do pipe é sempre zero — ele não grava dados, apenas liga dois processos como o | do shell. Entretanto, como o pipe tem um nome, os dois processos não precisam estar na mesma linha de comando ou mesmo serem executados pelo mesmo usuário.

Você pode testar isso fazendo o seguinte:

```
$ cd; mkfifo meupipe
$ echo "olá" >meupipe & # põe em plano de fundo (background)
[1] 5952
$ ls -l meupipe
prw-r--r--    1 pinguim pinguim  0 2003-11-06 23:18 meupipe
$ cat meupipe
hello
[1]+  Done                    echo olá >meupipe
$ ls meupipe
prw-r--r--    1 pinguim pinguim  0 2003-11-06 23:20 meupipe
$ rm mypipe
```

4.5.7 Soquetes

O soquete é similar ao pipe nomeado (FIFO) e possibilita que processos troquem informações. Para o soquete, esses processos não precisam ser executados ao mesmo tempo nem ser filhos do mesmo processo pai. Esse é o ponto final para a comunicação entre processos. A troca de informação pode ocorrer pela rede entre diferentes sistemas.

4.5.8 Arquivos de dispositivos

Arquivos de dispositivos se referem a dispositivos físicos ou virtuais em seu sistema, tais como seu disco rígido, sua placa de vídeo, monitor, ou teclado. Um exemplo de dispositivo virtual é o teclado, representado pelo `/dev/console`.

Há dois tipos de dispositivos:

- **dispositivo caracter**
  - Pode ser acessado um caracter por vez, isto é, a menor unidade de dado que pode ser escrita ou lida do dispositivo é um caracter (byte).
- **dispositivo bloco**
  - Precisa ser acessado em unidades maiores chamadas blocos, que contêm um número de caracteres. Seu disco rígido é um dispositivo bloco.

Você pode ler ou escrever em arquivos de dispositivos, embora o arquivo possa conter dados binários que podem ser algo sem nexo, incompreensível para humanos. Escrever dados diretamente nesses arquivos algumas vezes é útil para resolução de problemas de conexões de hardware. Por exemplo, você pode enviar um arquivo texto para o dispositivo de impressora `/dev/lp0` ou enviar comandos de modem para a porta serial apropriada `/dev/ttyS0`. Mas, a menos que isso seja feito com cautela, isso pode causar um grande desastre. Então seja cuidadoso.

`/dev/null` etc.

`/dev/null` é um arquivo de dispositivo especial que descarta qualquer coisa quq você escrever nele. Se você não quiser alguma coisa, envie para `/dev/null`. É essencialmente um poço sem fundo. Se você ler de `/dev/null`, obterá um caracter fim-de-arquivo (EOF) imediatamente.

`/dev/zero` é similar, mas se você ler a partir dele obtém o caracter `\0` (não o mesmo que o número zero ASCII). Veja 'Arquivos Falsos' on page 131.

Número de nó de dispositivo

O número de nó de dispositivo é mostrado executando `ls` assim:

```
$ ls -l /dev/hda /dev/ttyS0 /dev/zero
brw-rw----    1 root     disk       3,   0 Mar 14  2002 /dev/hda
```

```
crw-rw----    1 root     dialout    4,  64 Nov 15 09:51 /dev/ttyS0
crw-rw-rw-    1 root     root       1,   5 Aug 31 03:03 /dev/zero
```

Aqui,

- `/dev/hda` tem o número maior (major) de dispositivo 3 e número menor (minor) de dispositivo 0. É acessível para leitura/escrita pelo usuário que pertence ao grupo `disk`,
- `/dev/ttyS0` tem número maior de dispositivo 4 e número menor de dispositivo 64. É acessível para leitura/escrita pelo usuário que pertence ao grupo `dialout`, e
- `/dev/zero` tem número maior de dispositivo 1 e número menor de dispositivo 5. É acessível para leitura/escrita por qualquer um.

No sistema mais antigo, o processo de instalação cria os nós de dispositivos usando o comando `/sbin/MAKEDEV`. Veja a página de manual `MAKEDEV(8)`.

No sistema mais novo, o sistema de arquivos sob o `/dev` é povoado automaticamente pelo sistema de arquivos de dispositivos de forma similar ao sistema de arquivos `/proc`.

4.5.9 Sistema de arquivo `/proc`

O sistema de arquivos `/proc` é um pseudo sistema de arquivos e contém informações sobre o sistema e processos que estão sendo executados.

As pessoas freqüentemente se apavoram quando notam um arquivo em particular - `/proc/kcore` - que é geralmente enorme. Ele é (mais ou menos) uma cópia do conteúdo da memória de seu computador. É usado para depurar (debug) o kernel. Ele não existe realmente em lugar nenhum, então não se preocupe com seu tamanho.

Veja 'Ajustando o kernel pelo sistema de arquivos proc' on page 107 e a página de manual `proc(5)`.

4.6 Sistema X Window

Veja 'X' on page 147.

4.6.1 Iniciar o Sistema X Window

O Sistema X Window pode ser iniciado automaticamente com um daemon de login gráfico similar ao `xdm` ou digitando o seguinte a partir do console.

```
$ exec startx
```

4.6.2 Menu no Sistema X Window

Como o ambiente X pode acomodar muitos gerenciadores de janelas, suas interfaces de usuário variam bastante. Por favor, lembre-se de que clicando com o botão direito no plano de fundo (root window), obtem-se um menu de seleções. Isso está sempre disponível.

- Para obter o prompt de comando shell, inicie o Xterm a partir do menu:
  - "XShells" –> "XTerm".
- Para navegação gráfica de páginas web, inicie o Mozilla a partir do menu:
  - "Apps" –> "Net" –> "Mozilla Navigator".
- Para navegação gráfica de arquivos PDF, inicie o Xpdf a partir do menu:
  - "Apps" –> "Viewers" –> "Xpdf".

Se você não encontrar a entrada de menu, instale os pacotes pertinentes. Veja 'Básico do gerenciamento de pacotes Debian' on page 82.

4.6.3 Seqüência de teclado para o Sistema X Window

A seguir são apresentados os comandos de teclado importantes a lembrar ao executar o Sistema X Window.

- Ctrl-Alt-F1 até F6: Muda para outros pseudo-terminais (a partir de uma janela X, DOSEMU, etc.)
- Alt-F7: Volta para o X Window
- Ctrl-Alt-menos: Muda a resolução da tela no X Window (menos refere-se às teclas no teclado numérico)
- Ctrl-Alt-mais: Muda a resolução da tela da maneira oposta no X Window (mais refere-se às teclas no teclado numérico)
- Ctrl-Alt-Backspace: Termina o programa do Servidor X
- Alt-X, Alt-C, Alt-V: Combinações de teclas comuns do Windows/Mac para Cortar, Copiar e Colar com teclas Ctrl- são substituídas por essas teclas Alt- em alguns programas como o Netscape Composer.

4.7 Estudos adicionais

Nesse momento, recomendo que você leia os livros guia chaves a partir de O Projeto de Documentação Linux: Guias (`http://www.tldp.org/guides.html`):

- "The Linux System Administrators' Guide" (O Guia dos Administradores de Sistemas Linux),
  - Cobre todos os aspectos de como manter um sistema funcionando, manipulação de contas de usuários, backups, configuração do sistema.
  - pacote: `sysadmin-guide`
  - arquivo: `/usr/share/doc/sysadmin-guide/html/index.html`
  - web: `http://www.tldp.org/LDP/sag/index.html`
- "The Linux Network Administrator's Guide, Second Edition" (O Guia dos Administradores de Rede do Linux, Segunda Edição),

  - É uma referência única para administração de rede em um ambiente Linux.
  - pacote: `(not available)`
  - arquivo: `(notapplicable)`
  - web: `http://www.tldp.org/LDP/nag2/index.html`
- "Linux: Rute User's Tutorial and Exposition"
  - A nice online and hardcover book covering GNU/Linux system administration.
  - By Paul Sheer
  - Published by Prentice Hall
  - Package: `rutebook` (from `non-free`)
  - File: `/usr/share/doc/rutebook/`

Veja 'Suporte para o Debian' on page 249 para maiores recursos de aprendizado.

Capítulo 5

Atualizando uma distribuição para a `testing`

As notas de lançamento oficiais para atualização estão localizadas em `http://www.debian.org/releases/stable/releasenotes` e `http://www.debian.org/releases/testing/releasenotes` (trabalho em progresso).

O processo para atualizar o sistema para os sabores `testing/unstable` é:

- atualizar o sistema APT para a versão Woody se seu sistema for Potato, já que o APT do Potato não tinha as características descritas na página de manual `apt_preferences(5)` do Woody.
- modificar seus arquivos `/etc/apt/sources.list` e `/etc/apt/preferences`, para incluir referências à seção "testing" dos repositórios que você usa. Se você desejar, pode adicionar referências para a seção "unstable" também.
- atualizar suas listas de pacotes, e instalar quaisquer pacotes que agora são atualizáveis.

5.1 Transição do APT para a versão Woody

Atualização pela rede do sistema APT e de alguns pacotes principais para a versão Woody pode ser feita como a seguir depois de incluir as fontes `stable` no arquivo `/etc/apt/sources.list` se você ainda usar o Potato.

```
# apt-get update
# apt-get install libc6 perl libdb2 debconf
# apt-get install apt apt-utils dselect dpkg
```

5.2 Preparação para a transição (`stable` para `testing`)

Seguir o sabor `testing` do Debian tem o efeito colateral de fornecer atualizações de segurança muito lentamente. Então fique alerta.

Atualização pela rede para `testing` pode ser feita como explicado a seguir (execute o script go-woody (`http://www.debian.org/doc/manuals/debian-reference/examples/`) para fazer isso em um comando):

Esvazie o arquivo `sources.list` existente:

```
# cd /etc/apt
# cp -f sources.list sources.old
# :>sources.list
```

Obtenha uma lista limpa de repositórios, para `stable`:

```
# cd /
# apt-setup noprobe
 ... selecione repositórios, acessados com métodos HTTP ou FTP
```

Adicione a seção `testing` a essa nova lista. As linhas `deb-src` são comentadas.

```
# cd /etc/apt
# grep -e "^deb " sources.list  >sources.deb
# grep -e "^deb-" sources.list  >sources.src
# sed -e "s/stable/testing/"  sources.deb \
    >>sources.list
# sed -e "s/stable/testing/" sources.src | \
    sed -e "s/^deb-/#deb-/"  >>sources.list

# apt-get update
# apt-get install apt apt-utils
# cat >preferences <<EOF
 Package: *
 Pin: release a=testing
 Pin-Priority: 600

 Package: *
 Pin: release a=unstable
 Pin-Priority: 50

 EOF
```

Opcionalmente, adicione a seção “unstable” dos repositórios.

```
# sed -e "s/stable/unstable/" sources.deb \
     >>sources.list
# sed -e "s/stable/unstable/" sources.src | \
     sed -e "s/^deb-/#deb-/"  >>sources.list
```

Veja 'Básico do gerenciamento de pacotes Debian' on page 82 para conhecer a arte do ajuste do `/etc/apt/sources.list` e do `/etc/apt/preferences`.

Agora você pode atualizar a lista de pacotes e atualizar pacotes, usando métodos descritos na próxima seção.

5.3 Atualizar o sistema Debian

Depois de configurar os arquivos `/etc/apt/sources.list` e `/etc/apt/preferences` apropriadamente, o sistema pode ser atualizado para `testing`. Consulte 'Gerenciamento de pacotes Debian' on page 81 para saber o básico, e veja 'Resolvendo problemas com atualização pelo APT' on page 88 se encontrar problemas.

5.3.1 Melhor prática de atualização usando o `dselect`

Se um sistema tem muitos pacotes que incluem pacotes `-dev`, etc., recomenda-se o método a seguir usando o `dselect` para um controle de pacotes mais refinado.

```
# dselect update   # sempre faça isso antes de atualizar
# dselect select   # selecione pacotes adicionais
```

Todos os seus pacotes atuais serão selecionados quando o `dselect` iniciar. O `dselect` pode oferecer-lhe a opção de instalar pacotes adicionais, baseado em `depends`, `suggests`, e `recommends`. Se você não quiser adicionar nenhum pacote, simplesmente pressione `Q` para sair do `dselect` novamente.

```
# dselect install
```

Você precisará responder algumas questões de configuração de pacotes durante essa parte do processo, então mantenha suas notas prontas e reserve algum tempo a essa parte. Consulte '`dselect`' on page 83.

Use o `dselect`. **Ele sempre funciona :)**

5.3.2 Prática ruim de atualização com o `apt-get`

O uso do `apt-get` descrito abaixo é muito comum, mas *não* é recomendado para atualizações de sistema. Se você precisar atualizar sem o `dselect` depois do Woody, considere usar o `aptitude` e outras opções.

Se um sistema não tem muitos pacotes ou o repositório do Debian não teve grandes alterações, o método a seguir deve ser suficiente (algumas vezes).

```
# apt-get update # sempre faça isso antes de atualizar
 ... para atualizar o sistema com seleções "depends":
# apt-get upgrade                  # sempre faça isso antes de atualizar
 ... para atualizar o sistema todo com seleções "depends" :
# apt-get -u dist-upgrade
 ... ou para atualizar e manter com as seleções atuais do dselect (novo,
 melhor):
# apt-get -u dselect-upgrade       # usa o resultado do ajuste do dselect
```

Uma vez que esse método de atualização usa o `apt-get`, sua gestão de *recommends* e *suggests* é limitada. Veja 'Dependências de pacotes' on page 15.

Capítulo 6

Gerenciamento de pacotes Debian

Este capítulo é baseado em uma versão mais velha do texto original inglês. Verificar por favor a versão inglesa demasiado.

Para reduzir o tráfego de rede nos repositórios Debian, certifique-se de configurar um proxy HTTP local usando o `squid` para os pacotes os quais o download tenha sido feito através do APT e, caso necessário, define a variável de ambiente `http_proxy` ou defina o valor `http` no arquivo `/etc/apt/sources.list`. Isso melhora consideravelmente a performance de atualizações em rede, especialmente em diversas máquinas Debian em uma LAN (rede local).

Apesar do recurso de pinagem do `apt_preferences(5)` ser poderoso, ele não resolve todos os problemas de dependências uma vez que os requerimentos das dependências tendem a puxar versões mais novas de pacotes de programas fundamentais.

+ O uso do método descrito em '`chroot`' on page 131 é desejável para assegurar simultaneamente tanto a estabilidade do sistema quanto o acesso às últimas versões de softwares.

Este capítulo é baseado em um sistema Woody mas a maioria da informação também se aplica a um sistema Potato (exceto para o `apt_perferences(5)` e tópicos relacionados ao arquivo `/etc/apt/preferences`).

6.1 Introdução

Se ler toda a documentação para desenvolvedor é muito para você, leia este capítulo primeiro e comece a desfrutar de toda a capacidade do Debian com `testing/unstable` :-)

6.1.1 Ferramentas principais

```
dselect   - ferramenta de gerenciamento de pacotes orientada a menu (nível topo)
dpkg      - instala pacote (centrado em arquivo-pacote)
apt-get   - instala pacote (centrado em repositório de pacotes, CLI APT)
tasksel   - instala tarefa (um conjunto de pacotes)
```

```
aptitude  - instala pacote (pacote & tarefa, APT ncurses)
deity     - APT ncurses alternativo
synaptic, gsynaptic - alternativas GUI APT
```

Essas não são ferramentas de mesmo nível. O `dselect` roda no topo do APT (os comandos de linha são `apt-get`) e `dpkg`.

O APT usa `/var/lib/apt/lists/*` para rastrear pacotes disponíveis enquanto que o `dpkg` usa `/var/lib/dpkg/available`. Se você instalou pacotes diretamente usando `apt-get` ou programas similares como o `aptitude`, certifique-se de atualizar o arquivo `/var/lib/dpkg/available` a partir do item de menu `[U]pdate` no `dselect` ou a partir do comando shell "`dselect update`" antes de executar `dselect select`, `tasksel`, ou `dpkg -l`.

Como dependências de pacotes, o `apt-get` automaticamente insere pacotes com **depends** mas deixa os pacotes com **recommends** e **suggests**, enquanto o `dselect` oferece controle guiado por menus sob as opções desses pacotes e lhe pergunta pela escolha de pacotes baseado em **depends**, **recommends**, e **suggests**. O `aptitude` oferece a opção de inserir todos os pacotes automaticamente baseado em **depends**, **recommends** e **suggests**. Veja 'Dependências de pacotes' on page 15.

6.1.2 Ferramentas úteis

```
apt-cache          - verifica o repositório de pacotes no cache local
dpkg-reconfigure   - reconfigura um pacote já instalado
                     (se ele usar debconf)
dpkg-source        - gerencia arquivo de pacote fonte
dpkg-buildpackage - automatiza a construção de um arquivo de pacote.
...
```

6.2 Básico do gerenciamento de pacotes Debian

Você pode instalar um conjunto de pacotes chamados *tarefa* ou instalar pacotes individuais ou atualizar o sistema usando ferramentas de gerenciamento de pacotes como as ferramentas citadas abaxio.

Veja também 'Dicas para instalação do Sistema Debian' on page 27, 'Atualizando uma distribuição para a `testing`' on page 77 e 'Editores de recuperação' on page 212.

6.2.1 Instalar *tarefa* com o `tasksel` ou `aptitude`

O `tasksel` é o **Instalador de Tarefas Debian**, que é oferecido como a opção "`simple`" durante o sistema de instalação.

Quando se precisa instalar uma função comum que requer vários pacotes, essa é a melhor maneira de fazê-lo. Execute os comandos como a seguir:

```
# dselect update
# tasksel
```

O `aptitude` também oferece acesso a tarefas seletivamente através de menus. Ele não somente lhe permite selecionar *tarefas* mas também lhe permite desmarcar para instalação pacotes que fazem parte de uma *tarefa* seletivamente atráves de um menu.

6.2.2 Configure o sistema APT

Se você tentar seguir um ambiente misto como descrito aqui, pode comumente chegar a alguns conflitos de dependências de pacotes. É uma boa idéia não misturar sabores. O que vem a seguir é para pessoas que desejam experimentar reconhecendo alguns riscos.

Para atualização seletiva seguindo a distribuição `testing`, o sistema APT (>Woody) precisa ser configurado como em 'Transição do APT para a versão Woody' on page 77 para usar o recurso `apt_preferences(5)`.

Primeiro, adicione os fontes para `stable`, `tyesting`, e `unstable` em seu arquivo `/etc/apt/sources.list`. Edite então o arquivo `/etc/apt/preferences` para definir o Pin-Priority apropriado. [1]

```
Package: *
Pin: release a=stable
Pin-Priority: 500

Package: *
Pin: release a=testing
Pin-Priority: 600

Package: *
Pin: release a=unstable
Pin-Priority: 50
```

6.2.3 `dselect`

Quando iniciado, o `dselect` automaticamente seleciona todos os pacotes "Requeridos", "Importantes" e "Padrão". No sistema Potato, alguns programas grandes como o teTeX e o Emacs se encaixavam nessas categorias e era melhor não instalá-los na instalação inicial manualmente desmarcando para instalação os mesmos (digitando '_'). No Woody, esses pacotes foram movidos para a categoria de pacotes "Opcional".

O `dselect` possui uma interface com o usuário de certa forma estranha. Existem quatro comandos ambíguos (Maiúsculas significa MAIÚSCULAS") :

---

[1] Eu mantive a descrição por aqui como está para manter a consistência com outras partes. Se você realmente está seguindo a `testing` ou `unstable`, você deve na verdade remover referências à `stable` em `/etc/apt/sources.list` e `/etc/apt/preferences`. Isso é porque a `testing` começa como uma cópia da `stable`.

Tecla-Chave	Ação
Q	Finalizar. Confirma a seleção atual e finaliza de qualquer forma. (sobrescreve dependências)
R	Reverter ! Eu não queria dizer isso.
D	Droga ! Não me importa com o que o dselect acha. Faça isso !
U	Marque todos com o estado sUgerido

Com `D` e `Q`, você pode selecionar seleções conflitantes a seu próprio risco. Lide com esses comandos com cuidado. Atualmente, o `dselect` é a ferrementa guiada por menus madura que oferece um controle refinado sobre as escolhas de **suggests** e **recommends**.

Adicione uma linha contendo a opção "expert" no arquivo `/etc/dpkg/dselect.cfg` para reduzir o barulho. Para uma máquina mais lenta, você pode executar o `dselect` em uma outra máquina mais rápida para encontrar os pacotes e pode usar `apt-get install` para instalá-los.

O `dselect` não fornece acesso a pacotes que não sejam os preferidos pelo Pin-Priority.

6.2.4 `aptitude`

O `aptitude` é um novo instalador de pacotes guiado por menus similar ao `dselect`. Ele pode também ser usado como um comando alternativo de linha de comando no lugar do `apt-get`. Consulte a página de manual do `aptitude(1)`.

O `aptitude` aceita comandos de uma única tecla, geralmente em letras minúsculas.

Tecla-Chave	Ação
F10	Menu
?	Ajuda para as teclas de comandos
u	Atualiza informações dos pacotes do repositório
g	Faz o download e instala os pacotes selecionados
q	Finaliza a tela atual e salva as mudanças
x	Finaliza a tela atual e descarta as mudanças
Enter	Visualiza informações sobre um pacote

O `aptitude` oferece a opção de fazer o download de todos os pacotes automaticamente baseado nos **depends**, **recommends**, e **suggests**. Você pode mudar esse comportamente escolhendo `F10 -> Opções -> Gerenciamento de Dependências` nos menus do aptitude.

O `aptitude` oferece acesso a todas as versões de um pacote.

6.2.5 Comandos `apt-cache` e `apt-get`

Seguindo a distribuição `testing` como descrito no exemplo acima, podemos gerenciar o sistema usando os seguintes comandos:

- `apt-get -u upgrade`

  Esse comando segue a distribuição `testing` e atualiza todos os pacotes no sistema enquanto instala suas dependências da distribuição `testing`;

- `apt-get -u dist-upgrade`

  Esse comando segue a distribuição `testing` e atualiza todos os pacotes do sistema enquanto instala e resolve suas dependências da distribuição `testing`.

- `apt-get -u dselect-upgrade`

  Esse comando segue a distribuição `testing` e atualiza todos os pacotes no sistema pela seleção do `dselect`.

- `apt-get -u install` *`pacote`*

  Esse comando instala o pacote *pacote* e suas depedências da distribuição `testing`.

- `apt-get -u install` *`pacote`*`/unstable`

  Esse comando instala o pacote *pacote* da distribuição `unstable` enquanto instala suas dependências da distribuição `testing`.

- `apt-get -u install -t unstable` *`pacote`*

  Esse comando instala o pacote *pacote* da distribuição `unstable` enquanto instala suas depedências também da distribuição `unstable` definindo o Pin-Priority de `unstable` para 990.

- `apt-cache policy` *`foo bar ...`*

  Esse comando checa o estado dos pacotes *foo bar* . . . .

- `apt-cache show` *`foo bar ...`* `| less`

  Esse comando checa as informações dos pacotes *foo bar* . . . .

- `apt-get install` *`foo=2.2.4-1`*

  Esse comando instala a versão específica *2.2.4-1* do pacote *foo*.

- `apt-get -u install` *`foo bar-`*

  Esse comando instala o pacote *foo* e remove o pacote *bar*.

- `apt-get remove` *`bar`*

  Esse comando remove o pacote *bar* mas mantém intactos os arquivos de configuração que tenham sido modificados/personalizados.

- `apt-get remove --purge` *`bar`*

  Esse comando remove totalmente o pacote *bar* com todos os arquivos de configuração modificados/personalizados.

Nos exemplos acima, o uso da opção `-u` acima assegura que o `apt-get` exibirá uma lista de todos os pacotes que serão atualizados antes de efetuar a ação. O trecho a a seguir define a opção `-u` como padrão:

```
$ cat >> /etc/apt/apt.conf << .
//Sempre exibe pacotes a serem atualizados (-u)
APT::Get::Show-Upgraded "true";
.
```

Use a opção `-s` para simular um atualização sem na verdade fazê-la.

6.2.6 Seguindo um tipo da distribuição Debian

Dependendo de suas preferências por um sabor do Debian a seguir, você pode mudar o exemplo de `/etc/apt/preferences` em 'Configure o sistema APT' on page 83 para se adequar a suas necessidades :

```
seguir a estável:            mude o Pin-Priority da testing para 50
seguir a testing:            mantenha as confgiurações como exibidas acima
seguir a testing(unstable): mude o Pin-Priority da unstable para 50
seguir a unstable(testing): mude o Pin-Priority da unstable para 800
```

Uma linha guia para a escolha do Pin-Priority é mover de cima para baixo na tabela acima conforme o tempo passa de um momento imediatamente após o lançamento de uma distribução para um momento do congelamento (freeze) para o próximo lançamento.

Atenção: seguir o tipo `testing` do Debian possui o efeito colateral de obter atualizações de segurança muito lentamente.

Caso você misture os tipos do Debian `testing` com `stable` ou `unstable` com `stable`, você provavelmente irá obter também muitos pacotes-chave não intencionalmente do `testing` ou `unstable` que poderão conter bugs. Portanto, esteja avisado sobre esse problema.

Exemplos do arquivo `/etc/apt/preferences` que prendem alguns pacotes chave para suas versões mais maduras enquanto seguem as versões menos maduras de outros pacotes não-essenciais estão disponíveis no subdiretórios de exemplos (`http://www.debian.org/doc/manuals/debian-reference/examples/`) com o nome de `preferences.testing` e `preferences.unstable`. Por outro lado, o arquivo `preferences.stable` força todos os pacotes a serem rebaixados para suas versões existentes na distribuição "estável" (um "downgrade" na verdade).

6.2.7 Rebaixando todos os pacotes para a `estável`

Para rebaixar todos os pacotes para `stable`, edite `/etc/apt/preferences` como a seguir:

```
Package: *
Pin: release a=stable
Pin-Priority: 1001
```

e execute "`apt-get upgrade`", o que força o rebaixamento devido ao Pin-priority > 1000. Fique atento, podem haver alguns pequenos problemas com dependências.

6.2.8 Visão geral do arquivo `/etc/apt/preferences`

No arquivo `/etc/apt/preferences` de exemplo apresentado em 'Configure o sistema APT' on page 83, "Package: *" na primeira linha significa que esta seção se aplica a todos os pacotes. Você pode usar um nome de pacote ao invés de "*" para especificar o Pin-Priority desse pacote.

Na segunda linha, "Pin: release a=stable" significa que o `apt-get` obtém o arquivo `Packages.gz` de um repositório contendo o arquivo `Relese` que contenha a linha "Archive: stable".

Na última linha "Pin-Priority: 500" especifica o Pin-Priority como 500.

Caso existam diversos pacotes com o mesmo nome, normalmente aquele com o valor de Pin-Priority mais alto é o que será instalado.

Os significados de Pin-Priority são :

- 1001 e superior: Prioridades de "downgrade" (rebaixamento de versões).

  O "downgrade" (rebaixamento de versão) do pacote é permitido para esta faixa de Pin-Priority.
- 100 a 1000: Prioridades "Standard" (Padrão).

  O "downgrade" do pacote não é permitido. Alguns valores chave de Pin-Priority :
  - 990: a prioridade definida pela opção `--target-release>` ou `-t` do `apt-get(8)`.
  - 500: a prioridade para todos os arquivos de pacotes padrão.
  - 100: a prioridade para os arquivos de pacotes atualmente instalados.
- 0 a 99: Prioridades não-automáticas. (Estas são usadas somente caso o pacote não esteja instalado e não exista nenhuma outra versão disponível.)
- menor que 0: A versão não é selecionada nunca.

Os mesmos efeitos de `--target-release` podem ser obtidos configurando o arquivo `/etc/apt/apt.conf` como abaixo :

```
# echo 'APT::Default-Release "testing";' >> /etc/apt/apt.conf
```

A opção de linha de comando `--target-release` e a configuração no arquivo `/etc/apt/apt.conf` têm prioridade à configuração no arquivo `/etc/apt/preferences`. Seja cuidadoso com essa interferência quando estiver lidando com o arquivo `/etc/apt/preferences`.

6.3 Comandos de sobrevivência do Debian

Com esse conhecimento, é possível viver uma vida de "atualizações" **eternas** :-)

6.3.1 Verificar bugs no Debian e procurar ajuda

Se você está tendo problemas sobre um pacote específico, tenha certeza de verificar esses sites antes de pedir ajuda ou antes de enviar um relatório de bug. (`lynx`, `links`, e `w3m` funcionam igualmente bem):

```
$ lynx http://bugs.debian.org/
$ lynx http://bugs.debian.org/nome-do-pacote  # se você sabe o nome do pacote
$ lynx http://bugs.debian.org/número-do-bug   # se você sabe o número do bug
```

Procure no Google (www.google.com) com palavras de busca incluindo "site:debian.org“.

Quando tiver dúvidas, leia o manual. Configure `CDPATH` como a seguir:

```
export CDPATH=.:/usr/local:/usr/share/doc
```

e digite

```
$ cd nome-do-pacote
$ pager README.Debian # case esse exista
$ mc
```

Mais recursos de suporte estão listados em 'Suporte para o Debian' on page 249.

6.3.2 Resolvendo problemas com atualização pelo APT

Podem ocorrer problemas de dependência ao atualizar na `unstable/testing`, como descrito em 'Atualizar o sistema Debian' on page 79. Na maior parte das vezes, isso ocorre porque um pacote que será atualizado tem uma nova dependência que é desconhecida. Esses problemas são resolvidos usando

```
# apt-get dist-upgrade
```

Se isso não resolver, então repita um dos seguintes até que o problema se resolva:

```
# apt-get upgrade -f        # continua atualização depois do erro
... ou
# apt-get dist-upgrade -f   # continua atualização distribuição depois do erro
```

Alguns scripts de atualização realmente quebrados podem causar um problema persistente. É melhor resolver esse tipo de situação inspecionando os scripts `/var/lib/dpkg/info/nome-do-pacote.{post,pre}{inst,rm}` do pacote problemático e então executar:

```
# dpkg --configure -a    # configura todos os pacotes parcialmente instalados
```

Se um script reclamar de um arquivo de configuração inexistente, procure o arquivo de configuração correspondente em `/etc`. Se existir um com uma extensão `.new` (ou algo similar), modifique-o (`mv`) para remover o sufixo.

Podem ocorrer problemas de dependência ao instalar em `unstable/testing`. Há maneiras de enganar dependências.

```
# apt-get install -f pacote # anula dependências quebradas
```

Um método alternativo para resolver essas situações é usar o pacote `equivs`. Veja `/usr/share/doc/equivs/README.Debian` e 'O pacote `equivs`' on page 98.

6.3.3 Resgate usando o `dpkg`

A recuperação de uma falha no `dselect` (APT) pode ser feita em um sistema realmente quebrado usando apenas o `dpkg` sem o APT:

```
# cd /var/cache/apt/archives
# dpkg -i libc6* libdb2* perl*
# dpkg -i apt* dpkg* debconf*
# dpkg -i *  # até que não ocorram erros
```

Se estiver faltando um pacote, pegue-o a partir dos sites espelhos (`http://www.debian.org/mirror/`) com:

```
# mc            # use "link FTP" apontando para o servidor FTP Debian
```

Recentemente, os pacotes atuais no servidor HTTP/FTP podem não ser localizados sob o clássico diretório `/dist`, mas sob o novo diretório `/pool`. (Veja 'O diretório `pool`' on page 9.)

Então instale com:

```
# dpkg -i /var/cache/apt/archives/arquivodepacote.deb
```

Para uma dependência quebrada, conserte-a ou use:

```
# dpkg --ignore-depends=pacote1,... -i arquivodopacote.deb
# dpkg --force-depends -i arquivodopacote.deb
# dpkg --force-depends --purge pacote
# dpkg --force-confmiss -i arquivodopacote.deb  # Instala arquivo de configuração
```

6.3.4 Recuperação dos dados de seleção de pacotes

Se o arquivo `/var/lib/dpkg/status` se corromper por alguma razão, o sistema Debian perde os dados de seleção de pacotes e é severamente prejudicado. Procure pelo antigo arquivo `/var/lib/dpkg/status` em `/var/lib/dpkg/status-old` ou `/var/backups/dpkg.status.*`.

Manter `/var/backups/` em uma partição separada pode ser uma boa idéia já que esse diretório contém muitos dados importantes do sistema.

Se o arquivo `/var/lib/dpkg/status` antigo não estiver disponível, você ainda pode recuperar a informação a partir dos diretórios em `/usr/share/doc/`.

```
# ls /usr/share/doc | \
  grep -v [A-Z] | \
  grep -v '^texmf$' | \
  grep -v '^debian$' | \
  awk '{print $1 " install"}' | \
  dpkg --set-selections
# dselect --expert # reinstalar o sistema, de-selecione conforme necessário
```

6.3.5 Resgate do sistema após quebrar `/var`

Como o diretório `/var` contém dados atualizados regularmente como correio, ele é propício a ser corrompido. Separá-lo em uma partição independente limita os riscos. Se acontecer um desastre, você pode ter que recriar o diretório `/var` para resgatar seu sistema Debian.

Obtenha um conteúdo esqueleto do diretório `/var` a partir de um sistema Debian mínimo funcional baseado na mesma versão Debian ou mais antiga, por exemplo `var.tar.gz (http://people.debian.org/~osamu/pub/)`, e coloque-o no diretório raiz do sistema quebrado. Então

```
# cd /
# mv var var-old      # se algum conteúdo útil foi deixado
# tar xvzf var.tar.gz # use o arquivo esqueleto do Woody
# aptitude            # ou dselect
```

Isso deve prover um sistema funcional. Você pode realizar a recuperação das seleções de pacotes usando a técnica descrita em 'Recuperação dos dados de seleção de pacotes' on this page. ([FIXME]: Esse procedimento precisa mais experimentos para verificar.)

6.3.6 Instalar um pacote em um sistema não iniciável

Inicie no Linux usando um disquete/CD de resgate do Debian ou uma partição alternativa em um sistema Linux com multi-inicialização. Consulte 'Iniciando o sistema' on page 109. Monte o sistema não iniciável em `/target` e use o modo de instalação chroot do `dpkg`.

```
# dpkg --root /target -i arquivodopacote.deb
```

Então configure e resolva os problemas.

Por outro lado, se a única coisa que impede de iniciar é um `lilo` quebrado, você pode iniciar usando um disco de resgate padrão do Debian. No prompt de boot, considerando que a partição de sua instalação Linux está em `/dev/hda12` e você quer o nível de execução (runlevel) 3, entre:

```
boot: rescue root=/dev/hda12 3
```

Então você terá iniciado em um sistema quase totalmente funcional com o kernel no disquete. (Podem ocorrer problemas menores devido à falta de recursos do kernel ou módulos.)

6.3.7 O que fazer se o comando `dpkg` estiver quebrado

Um `dpkg` pode tornar impossível instalar qualquer arquivo `.deb`. Um procedimento como o seguinte o ajudará a recuperar-se dessa situação. (Na primeira linha, você pode substituir "links" com o comando de seu browser favorito.)

```
$ links http://http.us.debian.org/debian/pool/main/d/dpkg/
  ... baixe o dpkg_versão_arquit.deb bom
$ su
password: *****
# ar x dpkg_versão_arquitetura.deb
# mv data.tar.gz /data.tar.gz
# cd /
# tar xzfv data.tar.gz
```

Para `i386`, `http://packages.debian.org/dpkg` também pode usado como a URL.

6.4 Comandos de nirvana do Debian

A **Iluminação** com esses comandos salvará uma pessoa da eterna agonia cármica do inferno da atualização e a levará a alcançar o **nirvana** do Debian. :-)

6.4.1 Informação sobre um arquivo

Para encontrar o pacote ao qual um arquivo particular pertence:

```
$ dpkg {-S|--search} padrão # procura pelo padrão nos pacotes instalados
$ wget http://ftp.us.debian.org/debian/dists/sarge/Contents-i386.gz
$ zgrep -e pattern Contents-i386.gz
             # procura o padrão-nome-de-arquivo no arquivo do debian
```

Ou use comandos especializados de pacote:

```
# apt-get install dlocate
$ dlocate nome-de-arquivo  # alternativa rápida ao dpkg -L e dpkg -S
...
# apt-get install auto-apt # ferramenta de instalação de pacotes sob demanda
# auto-apt update          # cria arquivo db para o auto-apt
$ auto-apt search padrão
                # procura pelo padrão em todos os pacotes, instalados ou não
```

6.4.2 Informação sobre um pacote

Procura e exibe informação a partir dos arquivos de pacotes. Certifique-se de apontar o APT para o(s) repositório(s) próprio(s) editando `/etc/apt/sources.list`. Se você deseja ver como pacotes na `testing/unstable` estão em comparação com os atualmente instalados, use `apt-cache policy`—muito legal.

```
# apt-get   check          # atualiza o cache e verifica pacotes quebrados
$ apt-cache search  padrão # procura pacote pela descrição do texto
$ apt-cache policy  pacote # prioridade de pacote/informação de dists
$ apt-cache show -a pacote # exibe descrição do pacote em todas dists
$ apt-cache showsrc pacote # exibe descrição de pacote fonte determinado
$ apt-cache showpkg pacote # informação do pacote para depuração
# dpkg  --audit|-C         # procura pacotes parcialmente instalados
$ dpkg {-s|--status} pacote ... # descrição de pacote instalado
$ dpkg -l pacote ...       # estado de pacote instalado (1 linha cada)
$ dpkg -L pacote ...       # lista nomes de arquivos instalados pelo pacote
```

O `apt-cache showsrc` não foi documentado na versão Woody mas funciona :)

Você também pode encontrar informação sobre pacote em (eu uso o `mc` para navegar entre eles):

```
/var/lib/apt/lists/*
/var/lib/dpkg/available
```

A comparação dos seguintes arquivos dá informação sobre o que exatamente aconteceu nas últimas sessões de instalação.

```
/var/lib/dpkg/status
/var/backups/dpkg.status*
```

6.4.3 Instalação com APT sem acompanhamento

Para uma instalação sem acompanhamento, adicione a seguinte linha em `/etc/apt/apt.conf`:

```
Dpkg::Options {"--force-confold";}
```

Isso é equivalente a executar `apt-get -q -y` *`nome-do-pacote`*. Como isso responde "yes" automaticamente a todas as perguntas, pode causar problemas, então use essa dica com cuidado. Veja `apt.conf(5)` e `dpkg(1)`.

Você pode configurar quaisquer pacotes particulares mais tarde seguindo 'Reconfigurar pacotes instalados' on this page.

6.4.4 Reconfigurar pacotes instalados

Use o seguinte para reconfigurar qualquer pacote já instalado.

```
# dpkg-reconfigure --priority=medium pacote [...]
# dpkg-reconfigure --all   # reconfigura todos os pacotes
# dpkg-reconfigure locales # gera qualquer locale extra
# dpkg-reconfigure --p=low xserver-xfree86 # reconfigura servidor X
```

Faça isso para o `debconf` se você quiser mudar o modo de diálogo do `debconf` permanentemente.

Alguns programas vêm com scripts de configuração especiais.

```
apt-setup     - cria /etc/apt/sources.list
install-mbr   - instala um gerenciador de Master Boot Record
tzconfig      - configura o fuso horário local
gpmconfig     - configura o daemon de mouse gpm
eximconfig    - configura o Exim (MTA)
texconfig     - configura o teTeX
apacheconfig  - configura o Apache (httpd)
cvsconfig     - configura o CVS
sndconfig     - configura o sistema de som
...
update-alternatives - define o comando padrão, por exemplo, vim como vi
update-rc.d         - gerenciamento de script de init System-V
update-menus        - sistema de menus Debian
...
```

6.4.5 Remover e eliminar (purge) pacotes

Remover um pacote mantendo sua configuração:

```
# apt-get remove pacote ...
# dpkg  --remove pacote ...
```

Remover um pacote e toda sua configuração:

```
# apt-get remove --purge pacote ...
# dpkg    --purge        pacote ...
```

6.4.6 Mantendo pacotes antigos

Por exemplo, o travamento de `libc6` e `libc6-dev` para o `dselect` e `apt-get -u upgrade` *`pacote`* pode ser feito como a seguir:

```
# echo -e "libc6 hold\nlibc6-dev hold" | dpkg --set-selections
```

`apt-get -u install` *`pacote`* não será impedido por este "hold". Para manter um pacote forçando um rebaixamento automático para `apt-get -u upgrade` *`pacote`* ou `apt-get -u dist-upgrade`, adicione o seguinte a `/etc/apt/preferences`:

```
Package: libc6
Pin: release a=stable
Pin-Priority: 2000
```

Aqui o campo "`Package:`" não pode usar entradas como "`libc6*`". Se você precisa manter todos os pacotes binários relacionados ao pacote fonte `glibc` em uma versão sincronizada, precisa listá-los explicitamente.

O seguinte listará pacotes em "hold":

```
dpkg --get-selections "*"|grep -e "hold$"
```

6.4.7 Sistema híbrido `stable/testing/unstable`

`apt-show-versions` pode listar versões de pacotes disponíveis por distribuição.

```
$ apt-show-versions | fgrep /testing | wc
... quantos pacotes você tem da testing
$ apt-show-versions -u
... lista de pacotes atualizáveis
$ apt-get install `apt-show-versions -u -b | fgrep /unstable`
... atualiza todos os pacotes unstable para suas versões mais novas
```

6.4.8 Descartar arquivos de pacote em cache

A instalação de pacotes com o APT deixa pacotes em cache em `/var/cache/apt/archives` e esses precisam ser excluídos.

```
# apt-get autoclean # remove apenas os arquivos de pacotes sem uso
# apt-get clean     # remove todos os arquivos de pacotes em cache
```

6.4.9 Gravar/copiar a configuração do sistema

Para fazer uma cópia local dos estados de seleção dos pacotes:

```
# debconf-get-selections > debconfsel.txt
# dpkg --get-selections "*" >minhas_seleções   # ou use \*
```

″`*`‟ faz `minhas_seleções` incluir entradas de pacotes para ″purge“ também.

Você pode transferir esse arquivo para outro computador, e instalá-lo lá com:

```
# dselect update
# debconf-set-selections < debconfsel.txt
# dpkg --set-selections <minhas_seleções
# apt-get -u dselect-upgrade    # ou dselect install
```

6.4.10 Portar um pacote para o sistema `stable`

Para atualizações parciais do sistema `stable`, reconstruir um pacote dentro de seu ambiente usando o pacote fonte é desejável. Isso evita atualizações muito grandes de pacotes devido a suas dependências. Primeiro, adicione as seguintes entradas ao `/etc/apt/sources.list`:

```
deb-src http://http.us.debian.org/debian testing \
 main contrib non-free
deb-src http://http.us.debian.org/debian unstable \
 main contrib non-free
```

Aqui cada entrada para `deb-src` está quebrada em duas linhas devido a restrições de impressão, mas a entrada real em `sources.list` deve consistir de uma única linha.

Então pegue a fonte e faça um pacote local:

```
$ apt-get update  # atualiza a lista de busca de pacotes fontes
$ apt-get source pacote
$ dpkg-source -x pacote.dsc
```

```
$ cd pacote-versão
  ... inspecione pacotes necessários (Build-depends no arquivo .dsc) e
      instale-os também.  Você também precisa do pacote "fakeroot".

$ dpkg-buildpackage -rfakeroot

  ...ou (sem assinatura)
$ dpkg-buildpackage -rfakeroot -us -uc # use "debsign" depois se necessário

  ...Então para instalar
$ su -c "dpkg -i arquivodopacote.deb"
```

Geralmente, é necessário instalar uns poucos pacotes com o sufixo "-dev" para satisfazer dependências de pacotes. O `debsign` está no pacote `devscripts`. O `auto-apt` pode facilitar a satisfação dessas dependências. O uso de `fakeroot` evita o uso desnecessário da conta root.

No Woody, esses problemas de dependências podem ser simplificados. Por exemplo, para compilar um pacote `pine` de apenas fonte:

```
# apt-get build-dep pine
# apt-get source -b pine
```

6.4.11 Repositório local de pacotes

Para criar um repositório local de pacotes que seja compatível com o APT e o sistema `dselect`, é necessário criar o arquivo `Packages` e os arquivos de pacotes precisam ser colocados em uma árvore de diretórios específica.

Pode-se fazer um repositório `deb` local similar a um repositório Debian oficial assim:

```
# apt-get install dpkg-dev
# cd /usr/local
# install -d pool # os pacotes físicos são localizados aqui
# install -d dists/unstable/main/binary-i386
# ls -1 pool | sed 's/_.*$/ prioridade/ seção/' | uniq > override
# editor override # ajuste prioridade e seção
# dpkg-scanpackages pool override /usr/local/ \
   > dists/unstable/main/binary-i386/Packages
# cat > dists/unstable/main/Release << EOF
Archive: unstable
Version: 3.0
Component: main
Origin: Local
Label: Local
Architecture: i386
```

```
EOF
# echo "deb file:/usr/local unstable main" \
   >> /etc/apt/sources.list
```

Alternativamente, pode-se fazer um repositório `deb` local curto e grosso:

```
# apt-get install dpkg-dev
# mkdir /usr/local/debian
# mv /algum/lugar/pacote.deb /usr/local/debian
# dpkg-scanpackages /usr/local/debian /dev/null | \
  gzip - > /usr/local/debian/Packages.gz
#  echo "deb file:/usr/local/debian ./" >> /etc/apt/sources.list
```

Esses repositórios podem ser acessados remotamente disponibilizando acesso a esses diretórios através de métodos HTTP ou FTP e mudando as entradas em `/etc/apt/sources.list` apropriadamente.

6.4.12 Converter ou instalar um pacote binário alien

O `alien` possibilita a conversão de pacotes binários providos em formatos de arquivo `rpm` da Redhat, `slp` da Stampede, `tgz` do Slackware, e `pkg` do Solaris em um pacote `deb` do Debian. Se você quiser usar um pacote de outra distribuição Linux ao invés do que tem instalado em seu sistema, você pode usar o `alien` para convertê-lo para seu formato de pacotes preferido e instalá-lo. O `alien` também suporta pacotes LSB.

6.4.13 Verificar arquivos de pacotes instalados

`debsums` possibilita a verificação de arquivos de pacotes instalados com checksums MD5. Alguns pacotes não têm checksums MD5 disponíveis. Uma solução temporária para administradores de sistema:

```
# cat >>/etc/apt/apt.conf.d/90debsums
DPkg::Post-Install-Pkgs {"xargs /usr/bin/debsums -sg";};
^D
```

por Joerg Wendland <`joergland@debian.org`> (não testado).

6.4.14 Otimizando o arquivo `sources.list`

Resumidamente, esforços para criar um arquivo `sources.list` otimizado não produziram melhoramentos significativos para mim a partir de um local nos Estados Unidos (**nota do tradutor:** estas são as palavras do autor desta seção). Eu manualmente escolho um site mais próximo usando o `apt-setup`.

O `apt-spy` cria um arquivo `sources.list` automaticamente, baseando-se na latência e largura de banda. O `netselect-apt` cria um arquivo `sources.list` mais completo, mas utiliza um método inferior para escolher o melhor espelho (comparação do tempo de ping).

```
# apt-get install apt-spy
# cd /etc/apt ; mv sources.list sources.list.org
# apt-spy -d testing -l sources.apt
```

6.5 Outras peculiaridades do Debian

6.5.1 O comando `dpkg-divert`

Desvios (divertions) de arquivos são uma maneira de forçar o `dpkg` a não instalar um arquivo em sua localização padrão, mas em uma localização **desviada**. Os **desvios** podem ser usados através dos scripts de pacotes Debian para desviar de um arquivo quando ele causa um conflito. Os administradores de sistema também podem usar um desvio para substituir o arquivo de configuração de um pacote, ou quando alguns arquivos (que não estão marcados como **conffiles**) precisam ser preservados pelo `dpkg`, ao instalar uma versão mais nova de um pacote que contém aqueles arquivos (veja 'Preservação da configuração local' on page 13).

```
# dpkg-divert [--add]  nomedoarquivo # adiciona "desvio"
# dpkg-divert --remove nomedoarquivo # remove "desvio"
```

Geralmente é uma boa idéia não usar o `dpkg-divert` a menos que ele seja absolutamente necessário.

6.5.2 O pacote `equivs`

Se você compilar um programa a partir da fonte, é melhor torná-lo um pacote local debianizado real (`*.deb`). Use `equivs` como último recurso.

```
Package: equivs
Priority: extra
Section: admin
Description: Engana dependências de pacote Debian
 Este é um pacote falso que pode ser usado para criar pacotes
 Debian, que contém apenas informação de dependência.
```

6.5.3 Comandos alternativos

Para fazer o comando `vi` executar `vim`, use `update-alternatives`:

```
# update-alternatives --display vi
...
# update-alternatives --config vi
  Selection    Command
-----------------------------------------------
      1        /usr/bin/elvis-tiny
      2        /usr/bin/vim
*+    3        /usr/bin/nvi

Enter to keep the default[*], or type selection number: 2
```

Os itens no sistema de alternativas do Debian são mantidos em `/etc/alternatives` como ligações simbólicas.

Para definir seu ambiente X Window favorito, aplique `update-alternatives` para `/usr/bin/x-session-manager` e `/usr/bin/x-window-manager`. Para detalhes, veja 'Sessões X personalizadas' on page 153.

`/bin/sh` é uma ligação simbólica direta para `/bin/bash` ou `/bin/dash`. É mais seguro usar `/bin/bash` para ser compatível com scripts contaminados com velho Bashismo, mas é mais disciplinado usar `/bin/dash` para reforçar a compatibilidade POSIX. A atualização para um kernel Linux 2.4 tende a definir isso para `/bin/dash`.

6.5.4 `init` System-V e níveis de execução (runlevels)

O nível de execução padrão em que inicializar pode ser definido em `/etc/inittab`.

Diferente de outras distribuições, o Debian deixa o gerenciamento de nível de execução completamente como responsabilidade do administrador do sistema. O gerenciamento de `init` estilo System-V no Debian deve ser feito através de scripts `update-rc.d`.

Iniciar `/etc/init.d/`*`nome`* no nível de execução 1,2,3 e parar em 4,5 com número de prioridade de seqüência 20 (normal) pode ser feito com:

```
# update-rc.d nome start 20 1 2 3 . stop 20 4 5 .
```

A remoção de ligações simbólicas enquanto o script em `init.d` ainda existe pode ser feita com:

```
# update-rc.d -f nome remove
```

Para editar níveis de execução, eu trapaceio. Eu edito as entradas manualmente usando o comando `mv` no prompt shell do `mc` ao copiar as entradas de ligação usando Alt-Enter. Por exemplo:

```
# mv S99xdm K99xdm # desabilita xdm (X display manager)
```

Eu também desabilito o daemon inserindo `exit 0` no início de um script `init.d` como um hack rápido. Esses são os arquivos de configuração depois de tudo.

6.5.5 Serviços daemon desabilitados

A distribuição Debian considera a segurança do sistema seriamente e espera que o administrador do sistema seja competente. Assim, algumas vezes a facilidade de uso aparece como assunto secundário e muitos serviços daemon vêm com o maior nível de segurança, com os mínimos serviços (ou nenhum) disponíveis como seu estado de instalação padrão.

Execute `ps aux` ou verifique os conteúdos de `/etc/init.d/*` e `/etc/inetd.conf`, se tiver qualquer dúvida (sobre Exim, DHCP, . . .). Verifique também `/etc/hosts.deny` como em 'Restringindo logins com PAM' on page 139. O comando `pidof` também é útil (veja `pidof(8)`).

O X11 não permite conexões TCP/IP (remotas) por padrão nas versões recentes do Debian. Veja 'Usando o X sobre TCP/IP' on page 156. O redirecionamento do X em SSH também é desabilitado. Veja 'Conectando a um servidor X remoto – `ssh`' on page 157.

Capítulo 7

O kernel Linux no Debian

O Debian tem o seu próprio método de recompilar o kernel e módulos relacionados. Consulte também 'Debian e o kernel' on page 24.

7.1 Recompilação do Kernel

O uso do `gcc`, `binutils`, e `modutils` do Debian `unstable` pode ajudar quando compilamos o mais novo kernel Linux. Consulte `/usr/share/doc/kernel-package/README.gz`, especialmente o final do mesmo para a informação oficial.

Uma vez que trata-se de um alvo em movimento, a compilação do kernel é um assunto difícil que pode confundir até mesmo o desenvolvedor mais admirado:

Manoj Srivastava escreveu: `--initrd` requer um patch do cramfs somente para Debian.

Herbert Xu escreveu: Não, não precisa, tudo o que você precisa fazer para usar outro sistema de arquivos diferente do CRAMFS é definir MKIMAGE em `/etc/mkinitrd/mkinitrd.conf`.

Seja cuidadoso e sempre conte com `/usr/share/doc/kernel-package/README.gz` de Manoj e Kent. Certifique-se de obter a versão mais nova instável do pacote `kernel-package` caso esteja compilando a versão mais nova do kernel.

O `initrd` não é necessário para um kernel compilado somente para uma máquina. Eu o uso porque quero que meu kernel seja quase o mesmo que o kernel fornecido pelos pacotes kernel-image. Caso você use `initrd`, certifique-se de ler `mkinitrd(8)` e `mkinitrd.conf(5)`. Consulte também `http://bugs.debian.org/149236`.

7.1.1 Método Debian padrão

Fique atento aos relatórios de bugs do `kernel-package`, `gcc`, `binutils` e `modutils`. Use novas versões deles se necessário.

Compilar um kernel personalizado a partir dos fontes em um sistema Debian requer cuidado especial. Use o novo `--append_to_version` com o `make-kpkg` para construir múltiplas imagens do kernel.

```
# apt-get install debhelper modutils kernel-package libncurses5-dev
# apt-get install kernel-source-2.4.18   # Use a versão mais nova
# apt-get install fakeroot
# vi /etc/kernel-pkg.conf                # Coloque seu nome e e-mail
$ cd /usr/src                            # diretório de compilação
$ tar --bzip2 -xvf kernel-source-2.4.18.tar.bz2
$ cd kernel-source-2.4.18                # Caso esse seja seu fonte do kernel
$ cp /boot/config-2.4.18-386 .config     # use a configuração atual por padrão
$ make menuconfig                        # personalize-a como quiser
$ make-kpkg clean                        # deve ser executado (conforme: man make-
$ fakeroot make-kpkg --append_to_version -486 --initrd \
        --revision=rev.01 kernel_image \
        modules_image # modules_image é para pcmcia-cs*, etc.
$ cd ..
# dpkg -i kernel-image*.deb pcmcia-cs*.deb # instala
```

`make-kpkg kernel_image` na verdade faz um `make oldconfig` e um `make dep`. Não use `--initrd` caso não esteja usando um initrd.

Se quiser usar módulos de pcmcia-cs ou não usar pcmcia, deve-se selecionar ”General setup —>“ para ”PCMCIA/CardBus support —>“ no `make menuconfig` e definir a configuração para ”< > PCMCIA/CardBus support“ (ou seja, desmarque a caixa).

Em uma máquina SMP, defina CONCURRENCY_LEVEL de acordo com `kernel-pkg.conf(5)`.

7.1.2 Método clássico

Obtenha os fontes originais de :

- Linux: `http://www.kernel.org/`
- pcmcia-cs: `http://pcmcia-cs.sourceforge.net/`

ou use os fontes equivalentes no Debian e faça o seguinte :

```
# cd /usr/src
# tar xfvz linux-qualquercoisa.tar.gz
# rm -rf linux
# ln -s linux-qualquercoisa linux
# tar xfvz pcmcia-cs-qualquercoisa.tar.gz
# ln -s pcmcia-cs-qualquercoisa pcmcia
# cd linux
# make menuconfig
```

```
... coisas do configure ...
# make dep
# make bzImage
... edite para o lilo / grub ...
... mova /usr/src/linux/arch/i386/boot/bzImage para boot ...
... /sbin/lilo ou seja lá o que você faz para o grub
# make modules; make modules_install
# cd ../pcmcia
# make config
# make all
# make install
... adicione os nomes dos módulos necessários em /etc/modules
# shutdown -r now
... inicie com o novo kernel ...
```

7.1.3 Cabeçalhos do kernel

A maioria dos programas "normais" não precisa de cabeçalhos do kernel e, de fato, podem ter problemas se você usá-los diretamente; ao invés disso eles devem ser compilados com os mesmos cabeçalhos com os quais a **`glibc` foi compilada**, os quais são as versões em `/usr/include/linux` e `/usr/include/asm` do sistema Debian.

Portanto não coloque ligações simbólicas para os diretórios em `/usr/src/linux` de `/usr/include/linux` e `/usr/include/asm`, como sugerido por algumas documentações antigas.

Caso você **precise** de cabeçalhos de kernel particulares por alguns programas específicos de kernel, altere o(s) makefile(s) para que o(s) mesmo(s) inclua(m) caminhos apontando para *`diretório-dos-cabeçalhos-de-kernel-específicos`*`/include/linux` e *`diretório-dos-cabeçalhos-de-kernel-específicos`*`/include/asm`.

7.2 O kernel 2.4 modularizado

Os novos kernels Debian 2.4 fornecidos por `kernel-image-2.4.`*`NN`* são bem modulares. Você tem que se certificar que estes módulos estejam ativados para fazer com que o kernel funcione da maneira que você pretende.

Apesar de existirem muitos exemplos para `/etc/modules` na seção a seguir como uma rápida correção, eu ouvi que a maneira correta de corrigir estes problemas relacionados a módulos é fornecer um alias para o dispositivo em um arquivo em `/etc/modutils/` uma vez que existem aliases o bastante disponíveis com os kernels atuais. Alguns módulos podem ser auto ativados por programas de detecção de hardware como o `discover`. Veja também 'Detecção de hardware para o X' on page 149.

Consulte `Documentation/*.txt` nos fontes do kernel Linux para informação precisa.

7.2.1 PCMCIA

O arquivo `/etc/modules` pode precisar conter o conteúdo a seguir para que algum PCMCIA antigo funcione :

```
# driver ISA PnP
isa-pnp
# Novo Driver PCMCIA baixo-nível
# yenta_socket # não parece ser necessário em meu caso
```

O restante é gerenciado pelos scripts PCMCIA (do pacote `pcmcia-cs`), `depmod` e `kmod`. Eu acho que precisei do `isa-pnp` devido ao meu laptop ser um antigo ISA-PCMCIA. Laptops atuais com CardBus/PCMCIA podem não requerer isso.

Voz do gênio Miquel van Smoorenburg `<miquels@cistron.nl>`:

"Eu simplesmente removi todas as coisas pcmcia do laptop aqui no trabalho , incluindo o cardmgr, etc, e só instalei um kernel 2.4 com suporte cardbus e o novo pacote `hotplug` do woody.

Contanto que você possua somente cartões 32-bit você não precisa do pacote pcmcia; o kernel 2.4 possui o cardservices embutido. E o driver tulip padrão deve funcionar bem com seu cartão dlink.

—Mike."

Veja o HOWTO de PCMCIA no Linux (`http://www.tldp.org/HOWTO/PCMCIA-HOWTO.html`) e 'Configuração de rede e PCMCIA' on page 204.

7.2.2 SCSI

[NÃO TESTADO] O arquivo `/etc/modules` precisa conter o que se segue para que SCSI funcione :

```
# SCSI core
scsi_mod
# SCSI generic driver
sg
# SCSI disk
sd_mod
# Todos os outros módulos de HW necessários
...
```

O `depmod` pode cuidar de alguns dos módulos acima.

7.2.3 Função de rede

O arquivo `/etc/modules` precisa conter o que se segue para função extra de rede :

```
# net/ipv-4
ip_gre
ipip

# net/ipv-4/netfilter
# iptable (em ordem)
ip_tables
ip_conntrack
ip_conntrack_ftp
iptable_nat
iptable_filter
iptable_mangle
#
ip_nat_ftp
ip_queue
#
ipt_LOG
ipt_MARK
ipt_MASQUERADE
ipt_MIRROR
ipt_REDIRECT
ipt_REJECT
ipt_TCPMSS
ipt_TOS
ipt_limit
ipt_mac
ipt_mark
ipt_multiport
ipt_owner
ipt_state
ipt_tcpmss
ipt_tos
ipt_unclean
#
#ipchains
#ipfwadm
```

O precedente pode não estar otimizado. O `depmod` pode cuidar de alguns dos módulos acima.

7.2.4 Sistema de arquivos EXT3 (> 2.4.17)

Habilitar um sistema de arquivo com journaling com o FS EXT3 envolve os seguintes passos usando um pacote kernel-image (> 2.4.17) Debian pré-compilado :

```
# cd /etc; mv fstab fstab.old
# sed 's/ext2/ext3,ext2/g' <fstab.old >fstab
# vi /etc/fstab
... defina o tipo de sistema de arquivos do sistema de arquivos raiz para
... "auto" ao invés de "ext3,ext2"
# cd /etc/mkinitrd
# echo jbd >>modules
# echo ext3 >>modules
# echo ext2 >>modules
# cd /
# apt-get update; apt-get install kernel-image-2.4.17-686-smp
... instale o último kernel e configure a inicialização (lilo é executado aqui)
# tune2fs -j -i 0 /dev/hda1
# tune2fs -j -i 0 /dev/hda2
... Para todos os sistemas de arquivos EXT2 convertidos para EXT3
# shutdown -r now
```

Agora o journaling EXT3 está habilitado. Usar `ext3,ext2` como a entrada ″type“ do `fstab` garante um fallback seguro para EXT2 caso o kernel não suporte EXT3 para partições não-raíz.

Caso você tenha instalado um kernel 2.4 anteriormente e não deseja reinstalar, execute os passos acima até os comandos `apt-get` e então :

```
# mkinitrd -o /boot/initrd.img-2.4.17-686-smp /lib/modules/2.4.17-686-smp
# lilo
# tune2fs -j -i 0 /dev/hda1
# tune2fs -j -i 0 /dev/hda2
... para todos os sistemas de arquivos EXT2 convertidos para EXT3
# shutdown -r now
```

Agora journaling EXT3 está habilitado.

Caso `/etc/mkinitrd/modules` não estivesse configurado quando o `mkinitrd` foi executado e você quisesse adicionar alguns módulos em tempo de inicialização :

```
... no prompt initrd para ganhar o shell (5 segundos), digite RETURN
# insmod jbd
# insmod ext3 # modprobe ext3 pode cuidar de tudo
# insmod ext2
# ^D
... continue iniciando
```

Na tela de inicialização do sistema (`dmesg`), "cramfs: wrong magic" pode aparecer mas é conhecido que isso é inofensivo. Este problema foi resolvido no Sarge (2002/10). Consulte `http://bugs.debian.org/135537` e o mini-HOWTO do Sistema de Arquivos EXT3 (`http://www.zip.com.au/~akpm/linux/ext3/ext3-usage.html`) ou `/usr/share/doc/HOWTO/en-txt/mini/extra/ext3-mini-HOWTO.gz` para maiores informações.

Foi reportado que alguns sistemas experienciam severos travamentos de kernel caso EXT3 esteja habilitado mas eu não tive problemas (até o kernel 2.4.17).

7.2.5 Suporte Realtek RTL-8139 no kernel 2.4

Por alguma razão, o módulo para suporte a RTL-8139 não é mais chamado rtl8139, ele é agora chamado 8139too. Somente edite seu arquivo `/etc/modules` para refletir esta mudança quando atualizando um kernel 2.2 para um kernel 2.4.

7.2.6 Suporte a porta paralela

Para `kernel-image-2.4.*`, o suporte a porta paralela é fornecido como um módulo. Habilite-o usando :

```
# modprobe lp
# echo lp >> /etc/modules
```

Consulte `Documentation/parport.txt` nos fontes do kernel Linux.

7.3 Ajustando o kernel pelo sistema de arquivos proc

O comportamento do kernel Linux pode ser modificado durante a execução usando o sistema de arquivos proc.

Para obter informações básicas sobre modificação dos parâmetros do kernel através do sistema de arquivos `/proc`, leia `Documentation/sysctl/*` nos fontes do Linux.

Veja alguns exemplos de manipulações de parâmetros do kernel em `/etc/init.d/networking` and 'Estranhos problemas de acesso a alguns websites' on page 45.

Veja a página de manual `sysctl.conf(5)` para saber como como ajustar a configuração da inicialização do kernel através do sistema de arquivos `/proc` com o script`/etc/init.d/procps.sh` executado geralmente a partir de `/etc/rcS.d/S30procps.sh`.

7.3.1 Muitos arquivos abertos

O kernel Linux pode reclamar "Too many open files" (ou, "Muitos arquivos abertos"). Isto é devido ao pequeno valor padrão (8096) para `file-max`. Para corrigir esse problema, execute os comandos a seguir como root:

```
# echo "65536"  > /proc/sys/fs/file-max  # para kernel 2.2 e 2.4
# echo "131072" > /proc/sys/fs/inode-max # somente para kernel 2.2
```

ou coloque o seguinte em `/etc/sysctl.conf` para a mudança permanente:

```
file-max=65536   # para kernel 2.2 e 2.4
inode-max=131072 # somente para kernel 2.2
```

7.3.2 Intervalos de sincronização (flush) de disco

Você pode mudar os intervalos de sincronização (flush) do disco através do sistema de arquivos proc. O seguinte diminuirá esse intervalo do padrão de cinco segundos para um segundo.

```
# echo "40 0 0 0 100 30000 60 0 0"  > /proc/sys/vm/bdflush
```

Isso pode ter um pequeno impacto negativo no desempenho de I/O de arquivo. Mas isso assegura os conteúdos dos arquivos exceto para o último segundo, o que é mais curto que o padrão de cinco segundos. Isso é verdadeiro mesmo para o sistema de arquivos journaling.

7.3.3 Máquinas de pouca memória antigas e lerdas

Para alguns sistemas antigos de pouca memória, ainda pode ser útil habilitar sobre-envio (over-commit) de memória através do sistema de arquivos proc:

```
# echo 1 > /proc/sys/vm/overcommit_memory
```

Capítulo 8

Dicas Debian

8.1 Iniciando o sistema

Consulte o BootPrompt-HOWTO (`http://www.tldp.org/HOWTO/BootPrompt-HOWTO.html`) do LDP para obter informações detalhadas sobre o prompt de inicialização.

8.1.1 "Eu esqueci a senha do root ¡' (1)

É possível iniciar um sistema e logar com a conta do root sem conhecer a senha do root se você tiver acessso ao teclado do console. (Isto assume que não há requerimentos de senhas da BIOS e de um carregador de inicialização como o `lilo` que o impediriam de iniciar o sistema.)

Este é um procedimento que não requer discos de inicialização externos e nenhuma mudança nas configurações de inicialização. Aqui, "Linux" é o rótulo para inicializar o kernel Linux na instalação Debian padrão.

Na tela de inicialização do `lilo`, assim que `boot:` aparecer (você deve pressionar a tecla shift neste momento em alguns sistemas para prevenir a inicialização automática), digite :

```
boot: Linux init=/bin/sh
```

Isso faz com que o sistema inicie o kernel e execute `/bin/sh` ao invés de seu padrão `init`. Agora você ganhou privilégios de root e um shell de root. Uma vez que o `/` estará atualmente montado como somente-leitura e muitas partições de disco ainda não foram montadas, você deve fazer o seguinte para conseguir um sistema razoavelmente funcional.

```
init-2.03# mount -n -o remount,rw /
init-2.03# mount -avt nonfs,noproc,nosmbfs
init-2.03# cd /etc
init-2.03# vi passwd
init-2.03# vi shadow
```

(Caso o segundo campo de dados em `/etc/passwd` seja ″x‟ para cada nome de usuário, seu sistema usa senhas shadow e você deve editar o arquivo `/etc/shadow`.) Para desabilitar a senha do root, edite o segundo campo de dados no arquivo de senhas de maneira a deixá-lo vazio. Agora o sistema pode ser reiniciado e você pode logar como root sem uma senha. Quando inicia no nível de execução 1, o Debian (pelo menos depois do Potato) requer uma senha, o que algumas distribuições mais antigas não requerem.

É uma boa idéia ter um editor mínimo em `/bin/` caso o `/usr/` não esteja acessível (consulte 'Editores de recuperação' on page 212).

Considere também instalar o pacote `sash`. Quando o sistema ficar em um estado não-inicializável, execute :

```
boot: Linux init=/bin/sash
```

O `sash` funciona como um substituo interativo para o `sh` mesmo quando o `/bin/sh` está inutilizável. Ele é estaticamente ligado e inclui muitos utilitários padrões embutidos (tecle ″help‟ no prompt para uma lista de referência).

8.1.2 ″Eu esqueci a senha do root ¡' (2)

Inicie a partir de qualquer conjunto de disco de boot/root. Caso */dev/hda3* seja a partição raiz original, os comandos a seguir irão lhe permitir editar o arquivo de senhas tão facilmente quanto da maneira citada acima.

```
# mkdir fixit
# mount /dev/hda3 fixit
# cd fixit/etc
# vi shadow
# vi passwd
```

A vantagem deste método em relação ao método anterior é que você não precisa conhecer a senha do `lilo` (caso exista). Mas para usar este método você deve ser capaz de acessar o setup da BIOS e permitir o sistema iniciar a partir de um disquete ou de um CD, caso isto já não esteja configurado.

8.1.3 Não posso iniciar o sistema

Sem problemas, mesmo caso você não tenha se preocupado em criar um disco de inicialização durante a instalação. Caso o `lilo` esteja quebrado, pegue o disco de inicialização do conjunto de instalação Debian e inicie seu sistema a partir do mesmo. No prompt de inicialização, assumindo que a partição raiz de sua instalação Linux esteja em */dev/hda12* e você queira o nível de execução 3, digite :

```
boot: rescue root=/dev/hda12 3
```

Você irá iniciar então em um sistema quase que complementamente funcional usando o kernel no disquete. (Podem existir problemas menores devido a falta de recursos ou módulos do kernel.)

Veja também 'Instalar um pacote em um sistema não iniciável' on page 90 se tiver um sistema quebrado.

Caso você precise de um disquete de inicialização personalizado, siga as instruções de `readme.txt` no disco de recuperação.

8.1.4 "Deixe-me desabilitar o X na inicialização ¡'

Acompanhar a `unstable/sid` é divertido, mas uns `xdm`, `gdm`, `kdm` e `wdm` com bugs iniciados durante o processo de inicialização podem lhe deixar mal.

Primeiro obtenha o shell de root informando o seguinte no prompt de inicialização :

```
boot: Linux vga=normal s
```

Aqui, *Linux* é o rótulo para a imagem de kernel através da qual você está inicializando; "vga=normal" irá certificar-se de que o lilo execute em vídeo VGA comum e "s" (ou "S") é o parâmetro passado ao `init` para invocar o modo monousuário. Informe a senha de root no prompt.

Existem algumas maneiras de desabilitar todos os daemons X que iniciam :

- execute `update-rc.d ?dm stop 99 1 2 3 4 5 6 .`
- insira "exit 0" no início de todos os arquivos `/etc/init.d/?dm`.
- renomeie todos os arquivos `/etc/rc2.d/S99?dm` para `/etc/rc2.d/K99?dm`.
- remova todos os arquivos `/etc/rc2.d/S99?dm`.
- execute `:>/etc/X11/default-display-manager`

Aqui, o número em `rc2.d` deve corresponder ao nível de execução especificado em `/etc/inittab`. Também, `?dm` significa que você precisa executar o comando várias vezes substituindo-o com todos entre `xdm`, `gdm`, `kdm` e `wdm`.

Somente o primeiro é "a verdadeira maneira" no Debian. O último é fácil mas funciona somente no Debian e requer que você configure o display manager novamente depois usando `dpkg-reconfigure`. Os outros são métodos genéricos para desabilitar daemons.

Você ainda pode iniciar o X através do comando `startx` a partir de qualquer console shell.

8.1.5 Outros truques de inicialização com o prompt de inicialização

O sistema pode ser iniciado em um nível de execução e configuração particulares usando o prompt de inicialização do `lilo`. Detalhes são dados no BootPrompt-HOWTO (`http://www.tldp.org/HOWTO/BootPrompt-HOWTO.html`) (LDP).

Caso você queira iniciar o sistema no nível de execução 4, use o seguinte no prompt de inicialização do `lilo`.

```
boot: Linux 4
```

Caso você queira iniciar o sistema em modo monousuário funcionando normalmente e você conhece a senha de root, um dos exemplos a seguir no prompt de inicialização do `lilo` funcionará.

```
boot: Linux S
boot: Linux 1
boot: Linux -s
```

Caso você queira iniciar o sistema com uma quantidade de memória inferior a quantidade de memória que o sistema possui fisicamente (digamos, 48MB para um sistema com 64MB), use o seguinte no prompt de inicialização do `lilo` :

```
boot: Linux mem=48M
```

Certifique-se de não especificar mais do que a quantidade de memória que você possui fisicamente ou o kernel irá travar. Se você possui mais de 64MB de memória, por exemplo, 128MB, a menos que você execute `mem=128M` ou inclua uma linha adicional similar em `/etc/lilo.conf`, kernels antigos e/ou placas-mãe com uma BIOS antiga não utilizarão a memória acima de 64MB.

8.1.6 Definindo parâmetros de inicialização do GRUB

O GRUB é um novo gerenciador de inicialização do projeto Hurd e é muito mais flexível do que o Lilo, mas possui uma forma diferente de lidar com os parâmetros de inicialização.

```
grub> find /vmlinuz
grub> root (hd0,0)
grub> kernel /vmlinuz root=/dev/hda1
grub> initrd /initrd
grub> boot
```

Aqui, você deve conhecer os nomes de dispositivos Hurd :

```
o Hurd/GRUB          Linux                MSDOS/Windows
  (fd0)                /dev/fd0             A:
  (hd0,0)              /dev/hda1            C: (geralmente)
  (hd0,3)              /dev/hda4            F: (geralmente)
  (hd1,3)              /dev/hdb4            ?
```

Consulte `/usr/share/doc/grub/README.Debian.gz` e `/usr/share/doc/grub-doc/html/` para maiores detalhes.

8.2 Gravando atividades

8.2.1 Gravando atividades shell

Administração de sistema envolve tarefas muito mais elaboradas em um ambiente Unix do que em um computador pessoal comum. Certifique-se de conhecer os meios mais básicos de configuração caso você precise recuperar um sistema com problemas. Ferramentas de configuração baseados em X11 possuem uma boa aparência e são convenientes mas geralmente não podem ser utilizadas nessas situações de emergência.

Neste contexto, gravar atividades shell é uma boa prática, especialmente como root.

Emacs: Use M-x `shell` para iniciar gravando em um buffer, e use C-x C-w para gravar o conteúdo do buffer para um arquivo.

Shell: Use o comando `screen` com "^A H" como descrito em 'Mudança de console com `screen`' on page 128; ou use o comando `script`.

```
$ script
Script started, file is typescript
 ... faça qualquer coisa ...
 Ctrl-D
$ col -bx <typescript >savefile
$ vi savefile
```

O comando a seguir pode ser usado ao invés do `script` :

```
$ bash -i 2>&1 | tee typescript
```

8.2.2 Gravando atividades no X

Caso você precise gravar a imagem gráfica de uma aplicação X, incluindo um display xterm, use o `gimp` (GUI). Ele pode capturar cada janela ou a tela toda. Alternativas são o `xwd` (`xbase-clients`), `import` (`imagemagick`) e `scrot` (`scrot`).

8.3 Copiar e arquivar um subdiretório inteiro

8.3.1 Comandos básicos para copiar um subdiretório inteiro

Caso você precise rearranjar a estrutra de arquivos, mova o conteúdo incluindo ligações de arquivos usando :

```
Método padrão :
# cp -a /diretório/fonte /diretório/destino # requer o GNU cp
# (cd /diretório/fonte && tar cf - . ) | \
        (cd /diretório/destino && tar xvfp - )
Caso uma ligação direta (hard link) esteja envolvida, um método pedante é necessári
# cd /caminho/para/diretório/antigo
# find . -depth -print0 | afio -p -xv -0a /ponto/montagem/do/novo/diretório
Caso remoto:
# (cd /diretório/fonte && tar cf - . ) | \
        ssh usuário@máquina.domínio (cd /diretório/destino && tar xvfp - )
Caso não existam arquivos ligados :
# scp -pr usuário1@máquina1.domínio:/diretório/fonte \
          usuário2@máquina2.domínio:/diretório/destino
```

Aqui, `scp` <==> `rcp` e `ssh` <==> `rsh`.

A comparação informativa a seguir sobre copiar um subdiretório inteiro foi apresentada por Manoj Srivastava `<srivasta@debian.org>` na lista de discussão debian-user@lists.debian.org.

8.3.2 `cp`

Tradicionalmente, o `cp` não era um candidato para esta tarefa uma vez que ele não desreferenciava ligações simbólicas ou preservava ligações diretas (hard links). Outra coisa a considerar eram arquivos esparsos (arquivos com buracos).

O `cp` GNU superou estas limitações; porém, em um sistema não-GNU, o `cp` poderia continuar a ter problemas. Adicionalmente, você não pode gerar arquivos pequenos e portáveis usando o `cp`.

```
% cp -a . novodiretório
```

8.3.3 `tar`

O Tar superou alguns dos problemas que o `cp` tinha com ligações simbólicas. Porém, apesar do `cpio` lidar com arquivos especiais, o `tar` tradicional não o faz.

A maneira do `tar` lidar com mútiplas ligações diretas para um arquivo é colocar somente uma cópia da ligação na fita, mas o nome anexado a esta cópia é o *único* que você pode usar para recuperar o arquivo; a maneira do `cpio` é colocar uma cópia para cada ligação, mas você pode recuperá-lo usando quaisquer dos nomes.

O comando `tar` mudou sua opção para arquivos `.bz2` entre o Potato e o Woody, portanto, use `--bzip2` em scripts ao invés de sua forma abreviada `-I` (Potato) ou `-j` (Woody).

8.3.4 `pax`

O novo, cantante e dançante Utilitário de Troca de Arquivos Portável, compatível com os padrões POSIX (IEEE Std 1003.2-1992, páginas 380–388 (seção 4.48) e páginas 936–940 (seção E.4.48)). O `pax` irá ler, gravar e listar os membros de um arquivo e irá copiar hierarquias de diretórios. A operação do `pax` é independente do formato específico do arquivo e suporta uma grande variedade de formatos de arquivo.

As implementações do `pax` ainda são novas.

```
# apt-get install pax
$ pax -rw -p e . novodiretório
 ou
$ find . -depth  | pax -rw -p e  novodiretório
```

8.3.5 `cpio`

O `cpio` copia arquivos em ou de um arquivo `cpio` ou `tar`. O arquivo pode ser outro arquivo no disco, uma fita magnética ou um pipe.

```
$ find . -depth -print0 | cpio --null --sparse -pvd novo-diretório
```

8.3.6 `afio`

O `afio` é uma maneira melhor de lidar com arquivo no formato `cpio`. Ele é geralmente mais rápido do que o `cpio`, oferece opções de fita magnética mais diversas e lida de maneira graciosa com corrupção de dados de entrada. Ele suporta arquivo em múltiplos volumes durante a operação interativa. O `afio` pode criar arquivos comprimidos que são muito mais seguros do que arquivo comprimidos `tar` ou `cpio`. O `afio` é melhor usado como um "mecanismo de arquivo" em um script de backup.

```
$ find . -depth -print0 | afio -px -0a novo-diretório
```

Todos os meus backups em fita utilizam o `afio`.

8.4 Backups diferenciais e sincronização de dados

Backups diferenciais e sincronização de dados podem ser implementados com diversos métodos :

- `rcs`: backup e histórico, somente texto
- `rdiff-backup`: backup e histórico. ligação simbólica OK.
- `pdumpfs`: backup e histórico dentro de um sistema de arquivos. Ligação simbólica OK
- `rsync`: sincronização 1-way
- `unison`: sincronização 2-way
- `cvs`: sincronização multi-way com servidor backup e histórico, somente texto, maduro. Consulte 'Concurrent Versions System (CVS) (Sistema de Versões Concorrentes)' on page 217.
- `arch`: sincronização multi-way com servidor backup e histórico, não existe algo como um "diretório de trabalho".
- `subversion`: sincronização multi-way com servidor backup e histórico, Apache.

A combinação de um desses com o método descrito em 'Copiar e arquivar um subdiretório inteiro' on page 114 e a atividade regular automatizada descrita em 'Agendar atividade (`cron`, `at`)' on page 128 farão um bom sistema de backup.

Explicarei três utilitários de fácil uso.

8.4.1 Backup diferencial com rdiff

O `rdiff-backup` oferece backup legal e simples com histórico diferencial para quaisquer tipos de arquivos incluindo ligações simbólicas. Para fazer backup da maioria de `~/` para `/mnt/backup`:

```
$ rdiff-backup --include ~/tmp/keep --exclude ~/tmp  ~/ /mnt/backup
```

Para restaurar dados de três dias atrás do arquivo para `~/old`:

```
$ rdiff-backup -r 3D /mnt/backup ~/old
```

Consulte `rdiff-backup(1)`.

8.4.2 Backup diário com `pdumpfs`

O `pdumpfs` é um sistema de backup diário simples similar ao `dumpfs` do Plan9 que preserva todo estado diário. Você pode acessar estados (snapshots) antigos a qualquer momento para obter um arquivo de um certo dia. Vamos fazer um backup de seu diretório home com o `pdumpfs` e o `cron`!

O `pdumpfs` constrói o estado (snapshot) `YYYY/MM/DD` (ano/mês/dia) no diretório destino. Todos os arquivos fontes são copiados para o diretório do snapshot quando o `pdumpfs` é executado pela primeira vez. Na segunda vez e depois, o `pdumpfs` copia apenas os arquivos recém criados ou atualizados e grava os arquivos não modificados como ligações diretas para os arquivos do snapshot do dia anterior para economizar espaço em disco.

```
$ pdumpfs dir-fonte dir-dest [nomebase-dest]
```

Consulte `pdumpfs(8)`.

8.4.3 Backup diferencial regular com RCS

O `Changetrack` irá gravar regularmente as mudanças nos arquivos de configurações baseados em texto em arquivos RCS. Consulte `changetrack(1)`.

```
# apt-get install changetrack
# vi changetrack.conf
```

8.5 Recuperação de sistema paralizado

8.5.1 Matar um processo

Execute o `top` para ver qual processo está agindo de forma engraçada. Pressione 'P' para ordenar por uso de CPU, 'M' para ordenar por memória e 'k' para matar um processo. Alternativamente, o comando estilo BSD `ps aux | less` ou estilo System V `ps -efH | less` podem ser usados. A sintaxe estilo System V exibe o ID (`PPID`) do processo pai que pode ser usado para matar os filhos zumbis (defunct).

Use o `kill` para matar (ou enviar um sinal para) um processo pelo ID do processo, `killall` para fazer o mesmo por nome de comando do processo. Sinais freqüentemente usados :

```
 1: HUP,  reinicia daemon
15: TERM, kill normal
 9: KILL, kill forçado
```

8.5.2 Alt-SysRq

Garantia contra mal funcionamento do sistema é fornecida pela opção de compilação de kernel "Magic SysRq key". Pressionar Alt-SysRq em um i386, seguido por uma das teclas `r 0 k e i s u b`, faz a mágica.

Un'r'aw restaura o teclado após coisas como X travarem. Mudar o nível de log do console para '0' reduz mensagens de erros. sa'k' (tecla de administração de sistema) mata todos os processos no console virtual atual. t'e'rminate mata todos os processos no terminal atual exceto o `init`. k'i'll mata todos os processos exceto o `init`.

'S'ync, 'u'mount e re'b'oot são para se livrar de situações realmente ruins.

Os kernels de instalação padrão Debian não são compilados com essa opção no momento em que este documento é escrito. Recompile o kernel para ativar esta função. Informações detalhadas podem ser encontradas em `/usr/share/doc/kernel-doc-`*`versão`*`/Documentation/sysrq.txt.gz` ou `/usr/src/`*`kernel-versão`*`/Documentation/sysrq.txt.gz`.

8.6 Belos pequenos comandos para se lembrar

8.6.1 Paginador

O `less` é o paginador padrão (navegador de conteúdo de arquivo). Pressione 'h' para ajuda. Ele pode fazer muito mais do que o `more`. O `less` pode ser supercarregado executando `eval $(lesspipe)` ou `eval $(lessfile)` no script de inicialização do shell. Veja mais em `/usr/share/doc/lessf/LESSOPEN`. A opção `-R` permite saída de caracteres raw e habilita seqüências de escape de cores ANSI. Consulte `less(1)`.

O `w3m` pode ser um paginador alternativo útil para alguns sistemas de códigos (EUC).

8.6.2 Memória livre

O `free` e o `top` oferecem boa informação sobre recursos de memória. Não se preocupe sobre o tamanho de "used" na linha "Mem:", e ao invés leia a abaixo dessa (38792 no exemplo abaixo).

```
$ free -k # para uma máquina de 256MB
             total       used       free     shared    buffers cached
Mem:        257136     230456      26680      45736     116136 75528
-/+ buffers/cache:      38792     218344
Swap:       264996          0     264996
```

A quantidade exata de memória física pode ser confirmada com o comando `grep '^Memory' /var/log/dmesg`, que nesse caso exibe "Memory: 256984k/262144k available (1652k kernel code, 412k reserved, 2944k data, 152k init)".

```
Total              = 262144k = 256M (1k=1024, 1M=1024k)
Livre para o dmesg = 256984k = Total - kernel - reserved - data - init
Livre para o shell = 257136k = Total - kernel - reserved - data
```

Aproximadamente 5MB de memória não pode ser usada pelo sistema devido ao kernel utilizá-la.

8.6.3 Definir a hora (BIOS)

```
# date MMDDhhmmCCYY
# hwclock --utc --systohc
# hwclock --show
```

Isto irá definir a hora do sistema e de hardware para MM/DD hh:mm, CCYY. Horas são exibidas em tempo local mas a hora de hardware usa UTC.

8.6.4 Definir a hora (NTP)

Referência : HOWTO Gerenciando Data e Hora precisos (`http://www.tldp.org/HOWTO/TimePrecision-HOWTO/index.html`).

Definir a hora com conexão Internet permanente

Ajustar o relógio do sistema para a hora correta automaticamente através de um servidor remoto :

```
# ntpdate servidor
```

É bom ter esse comando em `/etc/cron.daily/` caso seu sistema possua uma conexão Internet permanente.

Definir a hora com uma conexão Internet esporádica

Use o pacote `chrony`.

8.6.5 Como controlar recursos de console como o protetor de tela

Para desabilitar o protetor de tela, use os seguintes comandos.

No console Linux :

```
# setterm -powersave off
```

Inicie o console kon2 (kanji) com :

```
# kon -SaveTime 0
```

Executando o X :

```
# xset s off
 ou
# xset -dpms
 ou
# xscreensaver-command -prefs
```

Leia as páginas de manual correspondentes para controlar outros recursos de console. Veja também a página de manual `stty(1)` para mudar e mostrar configurações de linha de terminal.

8.6.6 Procurar na base de dados administrativa

A Glibc oferece o `getent(1)` para procurar entradas nas bases de dados administrativas, isto é, passwd, group, hosts, services, protocols ou networks.

```
getent base_de_dados [chave ...]
```

8.6.7 Desabilitar o som (beep)

Sempre é possível desconectar o alto-falante do PC. ;-) Para o shell Bash:

```
echo "set bell-style none">> ~/.inputrc
```

8.6.8 Mensagens de erro na tela do console

Para acabar com as mensagens de erro na tela, o primeiro lugar a checar é `/etc/init.d/klogd`. Defina `KLOGD=″-c 3‶` nesse script e execute `/etc/init.d/klogd restart`. Um método alternativo é executar `dmesg -n3`.

Aqui os níveis de erro significam :

- 0: KERN_EMERG, o sistema está inutilizável
- 1: KERN_ALERT, uma ação deve ser tomada imediatamente
- 2: KERN_CRIT, condições críticas
- 3: KERN_ERR, condições de erro

- 4: KERN_WARNING, condições de aviso
- 5: KERN_NOTICE, normal mas em condições significantes
- 6: KERN_INFO, informativo
- 7: KERN_DEBUG, mensagens em nível de depuração

Caso uma mensagem de erro sem utilidade lhe incomode muito, considere fazer um patch para o kernel trivial como o `shutup-abit-bp6` (disponível no subdiretório de exemplos (`http://www.debian.org/doc/manuals/debian-reference/examples/`)).

Outro lugar a olhar pode ser o `/etc/syslog.conf`; cheque-o para ver se alguma mensagem é logada para um dispositivo console.

8.6.9 Definir o console para o tipo correto

Telas de console em sistemas semelhantes a Unix são normalmente acessadas usando as rotinas da biblioteca (n)curses. Estas fornecem ao usuário um método independente de terminal para atualizar caracteres de tela com razoável otimização. Consule `ncurses(3X)` e `terminfo(5)`.

Em um sistema Debian, existem uma porção de entradas pré-definidas :

```
$ toe | less                  # todas as entradas
$ toe /etc/terminfo/ | less   # entradas reconfiguráveis pelo usuário
```

Exporte sua seleção como a variável de ambiente `TERM`.

Caso a entrada terminfo para o `xterm` não funcione com um `xterm` não-Debian, mude seu tipo de terminal de ″xterm“ para uma das versões limitadas em recursos como ″xterm-r6“ quando você logar em um sistema Debian remotamente. Consulte `/usr/share/doc/libncurses5/FAQ` para maiores informações. ″dumb“ é o menor denominador comum para terminfo.

8.6.10 Voltar o console ao estador normal

Quando a tela fica cheia de caracteres estranhos depois de um comando `cat` *`algum-arquivo-binário`* (você pode não ser capaz de ver o comando digitado enquanto você digita) :

```
$ reset
```

8.6.11 Converter um arquivo texto DOS para o estilo Unix

Converter um arquivo texto DOS (fim-de-linha = `^M^J`) para um arquivo texto Unix (fim-de-linha = `^J`).

```
# apt-get install sysutils
$ dos2unix arquivo_dos
```

8.6.12 Substituição de expressões regulares

Substituir todas as instâncias de *DE_REGEX* por *PARA_TEXTO* em todos os arquivos *ARQUIVOS ...*:

```
$ perl -i -p -e 's/DE_REGEX/PARA_TEXTO/g;' ARQUIVOS ...
```

`-i` é para edição "in-place", `-p` é para "loop implícito sobre *ARQUIVOS ...*". Caso a substituição seja complexa, você pode tornar a recuperação de erros mais fácil usando o parâmetro `-i.bak` ao invés de `-i`; isso irá manter cada arquivo original, adicionando `.bak` como uma extensão de arquivo.

8.6.13 Editar arquivo no local usando um script

O script seguinte removerá as linhas 5–10 e linhas 16–20 no local.

```
#!/bin/bash
ed $1 <<EOF
16,20d
5,10d
w
q
EOF
```

Aqui, os comandos do `ed` são os mesmos comandos do command-mode do `vi`. Editar por trás do arquivo facilita a criação do script.

8.6.14 Extrair diferenças e juntar atualizações para os arquivos fonte

Seguir um desses procedimentos extrairá as diferenças entre dois arquivos fontes e criará arquivos diff unificados *arquivo.patch0* ou *arquivo.patch1* dependendo da localização dos arquivos:

```
$ diff -u arquivo.antigo arquivo.novo1 > arquivo.patch0
$ diff -u antigo/arquivo novo1/arquivo > arquivo.patch1
```

O arquivo diff (alternativamente chamado de arquivo patch) é usado para enviar atualizações de programas. O receptor aplicará esta atualização em outro *arquivo* usando :

```
$ patch -p0 arquivo < arquivo.patch0
$ patch -p1 arquivo < arquivo.patch1
```

Caso você possua três versões do código-fonte, você pode juntá-las mais efetivamente usando o `diff3`:

```
$ diff3 -m meu.arquivo arquivo.antigo seu.arquivo > arquivo
```

8.6.15 Converter um arquivo grande em arquivos pequenos

```
$ split -b 650m arquivo    # divide o arquivo em pedaços de 650MB
$ cat x* >arquivogrande    # junta arquivos em 1 grande arquivo
```

8.6.16 Extrair dados de uma tabela em arquivo texto

Vamos considerar um arquivo texto chamado `DPL` em que os nomes de todos os líderes anteriores do projeto Debian e seus dias de iniciação são listados em um formato separado por espaços.

```
Ian     Murdock   Agosto  1993
Bruce   Perens    Abril   1996
Ian     Jackson   Janeiro 1998
Wichert Akkerman  Janeiro 1999
Ben     Collins   Abril   2001
Bdale   Garbee    Abril   2002
Martin  Michlmayr Março   2003
```

O Awk é usado freqüentemente para extrair dados desses tipos de arquivo.

```
$ awk '{ print $3 }' <DPL                    # mês de início
Agosto
Abril
Janeiro
Janeiro
Abril
Abril
Março
$ awk '($1=="Ian") { print }' <DPL          # DPL chamado Ian
Ian     Murdock   Agosto  1993
Ian     Jackson   Janeiro 1998
$ awk '($2=="Perens") { print $3,$4 }' <DPL # Quando Perens começou
Abril 1996
```

Shells como o Bash também podem ser usados para interpretar esse tipo de arquivo:

```
$ while read primeiro ultimo mes ano; do
    echo $mes
  done <DPL
... mesma saída do primeiro exemplo Awk
```

Aqui, o comando interno (built-in) `read` usa os caracteres em $IFS (internal field separators) (separadores de campos internos) para dividir as linhas em palavras.

Se mudar IFS para ":", você pode interpretar o `/etc/passwd` com shell facilmente:

```
$ oldIFS="$IFS"   # salva o valor antigo
$ IFS=":"
$ while read usuario senha uid gid resto_da_linha; do
    if [ "$usuario" = "osamu" ]; then
      echo "O ID de $usuario é $uid"
    fi
  done < /etc/passwd
O ID de osamu é 1001
$ IFS="$oldIFS"  # restaura o valor antigo
```

(Se o Awk for usado para fazer o mesmo, use `FS=”:“` para definir o separador de campos.)

IFS também é usado pelo shell para dividir resultados de expansão de parâmetros, substituição de comandos, e expansão aritmética. Esses não ocorrem em palavras com aspas duplas ou simples (double or single quoted words). O valor padrão de IFS é <espaço>, <tab> e <nova_linha> combinados.

Tenha cuidado ao usar esses truques do IFS do shell. Coisas estranhas podem acontecer, quando o shell interpreta algumas partes do script como sua **entrada**.

```
$ IFS=":,"                           # usa ":" e "," como IFS
$ echo IFS=$IFS,   IFS="$IFS"        # echo é um comando interno (built-in) do Bash
IFS=  , IFS=:,
$ date -R                            # apenas uma saída de comando
Sat, 23 Aug 2003 08:30:15 +0200
$ echo $(date -R)                    # sub shell --> entrada para shell principal
Sat  23 Aug 2003 08 30 36 +0200
$ unset IFS                          # redefine IFS para o padrão
$ echo $(date -R)
Sat, 23 Aug 2003 08:30:50 +0200
```

8.6.17 Pequenos scripts para ligar comandos

Os scripts a seguir farão coisas legais como parte de um pipe.

```
find /usr | egrep -v "/usr/var|/usr/tmp|/usr/local"
                     # encontra todos os arquivo em /usr excluindo alguns arquivos
xargs -n 1 comando   # executa o comando para todos os itens de stdin
xargs -n 1 echo |    # divide itens separados-por-espaços-em-branco em linhas
xargs echo      |    # mescla todas as linhas em uma linha
grep -e padrão|      # extrai linhas contendo padrão
cut -d: -f3 -|
        # extrai o terceiro campo separado por : (arquivo passwd etc.)
awk '{ print $3 }' | # extrai o terceiro campo separado por espaços
awk -F'\t' '{ print $3 }' |
```

```
        # extrai terceiro campo separado por tab
col -bx |             # remove backspace e expande tabs para espaços
expand -|             # expande tabs
sort -u|              # ordena e remove duplicados

tr '\n' ' '|          # concatena linhas em uma linha
tr '\r' ''|           # remove CR
tr 'A-Z' 'a-z'|       # converte maiúsculas em minúsculas
sed 's/^/# /'|        # torna cada linha um comentário
sed 's/\.ext//g'|     # remove .ext
sed  -n -e 2p|        # imprime a segunda linha
head -n 2 -|          # imprime as primeiras 2 linhas
tail -n 2 -|          # imprime as últimas 2 linhas
```

8.6.18 Pequenos scripts para fazer loop para cada arquivo

As seguintes maneiras de fazer loop para cada arquivo do tipo `*.ext` asseguram a manipulação apropriada de nomes de arquivos engraçados como aqueles com espaços e executam processo equivalente:

- Loop Shell:

```
for x in *.ext; do
  if test -f "$x"; then
    comando "$x"
  fi
done
```

- Combinação de `find` e `xargs`:

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | \
 xargs -0 -n 1 comando
```

- `find` com a opção `-exec` com um comando:

```
find . -type f -maxdepth 1 -name '*.ext' \
 -exec comando '{}' \;
```

- `find` com a opção `-exec` com um script shell curto:

```
find . -type f -maxdepth 1 -name '*.ext' \
 -exec sh -c "comando '{}' && echo 'successful'" \;
```

8.6.19 Loucura de script Perl curto

Apesar de quaisquer scripts Awk poderem ser reescritos em Perl usando `a2p(1)`, scripts Awk de uma linha são melhor convertidos para scripts perl de uma linha manualmente. Por exemplo

```
awk '($2=="1957") { print $3 }' |
```

é equivalente a qualquer uma das seguintes linhas:

```
perl -ne '@f=split; if ($f[1] eq "1957") { print "$f[2]\n"}' |
perl -ne 'if ((@f=split)[1] eq "1957") { print "$f[2]\n"}' |
perl -ne '@f=split; print $f[2] if ( $f[1]==1957 )' |
perl -lane 'print $F[2] if $F[1] eq "1957"' |
```

Como todos os espaços nos argumentos para o `perl` na linha acima podem ser removidos, e tomando vantagem das conversões automáticas entre números e strings no Perl:

```
perl -lane 'print$F[2]if$F[1]eq+1957' |
```

Veja `perlrun(1)` para detalhes sobre opções de linha de comando. Se quiser mais scripts Perl loucos, `http://perlgolf.sourceforge.net` pode ser interessante.

8.6.20 Obter texto ou arquivo de lista de discussão de uma página web

O comando a seguir irá ler uma página web em um arquivo texto. Muito útil quando copiando configurações da Web.

```
$ lynx -dump http://www.remote-site.com/help-info.html >arquivotexto
```

O `links` e o `w3m` podem ser usados aqui também, com pequenas diferenças na renderização.

Caso esteja lidando com um arquivo de lista de discussão, use o `munpack` para obter conteúdos mime do texto.

8.6.21 Imprimir de forma bonita uma página web

O comando a seguir irá imprimir uma página Web em um arquivo/impressora Postscript :

```
$ apt-get install html2ps
$ html2ps URL | lpr
```

Consulte '`lpr/lpd`' on page 41. Confira também os pacotes `a2ps` e `mpage` para criar arquivos Postscript.

8.6.22 Imprimir de forma bonita uma página de manual

O comando a seguir irá imprimir uma página de manual em um arquivo/impressora Postscript :

```
$ man -Tps alguma-página-de-manual | lpr
$ man -Tps alguma-página-de-manual | mpage -2 | lpr
```

8.6.23 Juntar dois arquivos PostScript ou PDF

Você pode juntar arquivos PostScript ou PDF.

```
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pswrite \
  -sOutputFile=bla.ps -f foo1.ps foo2.ps
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pdfwrite \
  -sOutputFile=bla.pdf -f foo1.pdf foo2.pdf
```

8.6.24 Contar o tempo de um comando

Exibir o tempo usado por um processo.

```
# time algum-comando >/dev/null
real    0m0.035s       # tempo no relógio de parede (tempo real decorrido)
user    0m0.000s       # tempo em modo de usuário
sys     0m0.020s       # tempo em modo kernel
```

8.6.25 Comando `nice`

Use o `nice` (do pacote GNU `shellutils`) para definir um valor de nice de um comando quando iniciá-lo. O `renice` (`bsdutils`) ou o `top` podem fazer o renice de um processo. Um valor nice de 19 representa o processo mais lento (a menor prioridade); valores negativos são ″não-nice“, com -20 sendo um processo bem rápido (alta prioridade). Somente o superusuário pode definir valores de nice negativos.

```
# nice  -19 top                                      # muito "nice"
# nice --20 cdrecord -v -eject speed=2 dev=0,0 disk.img # muito rápido
```

Algumas vezes um valor de nice extremo traz mais problemas do que melhorias para o sistema. Use esse comando com cuidado.

8.6.26 Agendar atividade (`cron`, `at`)

Use o `cron` e o `at` para agendar tarefas sob o Linux. Consulte `at(1)`, `crontab(5)` e `crontab(8)`.

Execute o comando `crontab -e` para criar ou editar um arquivo crontab para configurar eventos regularmente agendados. Exemplo de um arquivo crontab :

```
# use /bin/sh para executar comandos, não importando o que o /etc/passwd diga
SHELL=/bin/sh
# envie por mail qualquer saída para `paul', não importando de quem seja esse cront
MAILTO=paul
# Min Hora DiaDoMês Mês DiaDaSemana comando (Dia... são OU)
# executa às 00:05, todos os dias
5  0  *  * *   $HOME/bin/daily.job >> $HOME/tmp/out 2>&1
# executa às 14:15 no primeiro dia de cada mês -- saída enviada para paul
15 14 1  * *   $HOME/bin/monthly
# executa às 22:00 em dias de semana(1-5), saída para Joe. % para nova linha, últi
0 22 *   * 1-5 mail -s "It's 10pm" joe%Joe,%%Onde estão suas crianças?%.%%
23 */2 1 2 *   echo "executa 23 minutos depois de 0am, 2am, 4am ..., em 1 de Fevere
5  4 *   * sun echo "executa às 04:05 todo domingo"
# executa às 03:40 na primeira Segunda-feira de cada mês
40 3 1-7 * *   [ "$(date +%a)" == "Mon" ] && comando -argumentos
```

Execute o comando `at` para agendar uma atividade a ser executada apenas uma vez :

```
$ echo 'comando -argumentos'| at 3:40 monday
```

8.6.27 Mudança de console com `screen`

O programa `screen` lhe permite executar **múltiplos** terminais virtuais, cada um com seu próprio shell interativo, em um **único** terminal físico ou janela de emulação de terminal. Mesmo caso você use consoles virtuais Linux ou múltiplas janelas xterm, vale a pena explorar o `screen` devido a seu abundante **conjunto de recursos**, os quais incluem

- histórico scrollback,
- copiar-e-colar,
- logging de saída,
- entrada dígrafo e
- a habilidade de **separar** uma sessão `screen` inteira de seu terminal e anexá-la posteriormente.

Cenários de acesso remoto

Caso você freqüentemente logue em uma máquina Linux de um terminal remoto ou usando um programa de terminal VT100, o `screen` irá tornar sua vida mais fácil com o recurso de **detach**.

1. Você está logado através de uma conexão discada e está executando uma sessão `screen` complexa com editores e outros programas abertos em diversas janelas.
2. Em certo momento você precisa deixar seu terminal, mas você não quer perder seu trabalho desconectando.
3. Simplesmente digite `^A d` para **separar** a sessão e então faça o logout. (Ou, ainda mais rápido, digite `^A DD` para fazer com que o `screen` separe e faça o logout dele mesmo.)
4. Quando você logar novamente posteriormente, digite o comando `screen -r` e o `screen` irá magicamente **anexar** todas as janelas que você deixou abertas.

Comandos `screen` típicos

Uma vez que você inicia o `screen`, toda a entrada de teclado é enviada para sua janela atual exceto as teclas de comando, por padrão `^A`. Todos os comandos `screen` são informados digitando `^A` mais uma única tecla [mais quaisquer parâmetros]. Comandos úteis :

^A ?	exibe uma tela de ajuda (exibe os mapeamentos de teclas)
^A c	cria uma nova janela e muda para ela
^A n	vai para a próxima janela
^A p	vai para a janela anterior
^A *0*	vai para a janela número *0*
^A w	exibe uma lista de janelas
^A a	envia um Ctrl-A para a janale atual como entrada de teclado
^A h	grava uma cópia "hard" da janela atual para arquivo
^A H	inicia/finaliza log da janela atual para arquivo
^A ^X	trava o terminal (protegido por senha)
^A d	separa sessão screen do terminal
^A DD	separa a sessão screen e faz um logout

Isto é somente um pequeno subconjunto dos comandos e recursos do `screen`. Caso exista alguma coisa que você queira que o `screen` seja capaz de fazer, as chances são que que possa fazê-lo ! Consulte `screen(1)` para maiores detalhes.

Backspace e/ou Ctrl-H em sessão `screen`

Se você achar que o backspace e/ou o Ctrl-H não funcionam corretamente quando você está executando o `screen`, edite `/etc/screenrc`, encontre a linha contendo

```
bindkey -k kb stuff "\177"
```

e comente-a (ou seja, adicione "#" como o primeiro caracter).

Programa equivalente ao `screen` para X

Confira o `xmove`. Consulte `xmove(1)`.

8.6.28 Testes de rede básicos

Instale os pacotes `netkit-ping`, `traceroute`, `dnsutils`, `ipchains` (para kernel 2.2), `iptables` (para kernel 2.4) e `net-tools` e :

```
$ ping yahoo.com              # checa a conexão Internet
$ traceroute yahoo.com        # rastrea pacotes IP
$ ifconfig                    # checa configuração do host
$ route -n                    # checa configuração de roteamento
$ dig [@servidor-dns.com] host.domínio [{a|mx|any}] |less
      # checa os registros DNS de host.domínio usando servidor-dns.com
      # para um registro a {mx|any}
$ ipchains  -L -n |less       # checa o filtro de pacotes (kernel 2.2)
$ iptables -L -n |less        # checa o filtro de pacotes (kernel 2.4)
$ netstat -a                  # encontra todas as portas abertas
$ netstat -l --inet           # encontra as portas em escuta
$ netstat -ln --tcp           # encontra todas as portas TCP em escuta (numérico)
```

8.6.29 Fazer flush em mensagens do spool local

Para fazer um flush das mensagens do spool local :

```
# exim -q    # faz flush em mensagens aguardando
# exim -qf   # faz flush em todas as mensagens
# exim -qff  # faz flush até mesmo em mensagens congeladas
```

`-qff` pode ser melhor como uma opção no script `/etc/ppp/ip-up.d/exim`.

8.6.30 Remover mensagens congeladas do spool local

Para remover mensagens congeladas do spool local com uma mensagem de erro de entrega :

```
# exim -Mg `mailq | grep frozen | awk '{ print $3 }'`
```

8.6.31 Reentregar conteúdo mbox

Você precisa entregar manualmente mensagens para caixas de mensagens ordenadas em seu diretório home a partir de `/var/mail/`*`username`* caso seu diretório home fique lotado e o procmail falhe. Depois de liberar espaço em disco no diretório home, execute :

```
# /etc/init.d/exim stop
# formail -s procmail </var/mail/nomedeusuário
# /etc/init.d/exim start
```

8.6.32 Limpar conteúdo de arquivo

Para limpar o conteúdo de um arquivo como um arquivo de log, não use o `rm` para apagar o arquivo e então criar um novo arquivo vazio porque o arquivo pode estar sendo acessado no intervalo entre os comandos. O comando a seguir é uma maneira mais segura de limpar o conteúdo de um arquivo :

```
$ :>arquivo-a-ser-limpo
```

8.6.33 Arquivos Falsos

Os comandos a seguir criarão arquivos falsos ou vazios :

```
$ dd if=/dev/zero    of=nomedearquivo bs=1k count=5 # 5KB de conteúdo zero
$ dd if=/dev/urandom of=nomedearquivo bs=1M count=7 # 7MB de conteúdo randômico
$ touch nomedearquivo #  cria arquivo de 0B (caso o arquivo exista, atualiza seu mt
```

Por exemplo, os seguintes comandos executados a partir do shell do disco de inicialização do Debian apagarão todo o conteúdo do disco rígido `/dev/hda` completamente para muitos usos práticos.

```
# dd if=/dev/urandom of=/dev/hda ; dd if=/dev/zero of=/dev/hda
```

8.6.34 `chroot`

O programa `chroot`, `chroot(8)`, nos possibilita executar diferentes instâncias do ambiente GNU/Linux em um único sistema simultaneamente sem reiniciar.

Pode-se também executar um programa que exija muitos recursos como o `apt-get` ou o `dselect` sob o chroot de uma máquina host rápida enquanto montando por NFS uma máquina satélite lenta no host como leitura/escrita e o ponto de chroot sendo o ponto de montagem da máquina satélite.

Executar um sabor diferente do Debian com `chroot`

Um ambiente Debian chroot pode ser criado facilmente pelo comando `debootstrap` no Woody. Por exemplo, para criar um chroot Sid em */sid-root* tendo acesso rápido à Internet:

```
main # cd / ; mkdir /sid-root
main # debootstrap sid /sid-root http://ftp.debian.org/debian/
... assista-o baixar o sistema inteiro
main # echo "proc-sid /sid-root/proc proc none 0 0" >> /etc/fstab
```

```
main # mount proc-sid /sid-root/proc -t proc
main # cp /etc/hosts /sid-root/etc/hosts
main # chroot /sid-root /bin/bash
chroot # cd /dev; /sbin/MAKEDEV generic ; cd -
chroot # apt-setup # configure /etc/apt/sources.list
chroot # vi /etc/apt/sources.list # aponte a fonte para unstable
chroot # dselect  # você pode usar aptitude, instalar o mc e o vim :-)
```

Nesse ponto você deve ter um sistema Debian completamente funcional, onde pode mexer sem medo de afetar sua instalação Debian principal.

Esse truque do `debootstrap` também pode ser usado para instalar o Debian em um sistema sem usar o disco de instalação do Debian, mas usando um de outra distribuição GNU/Linux. Veja `http://www.debian.org/releases/stable/i386/apcs04`.

Configurando o login para `chroot`

Digitar `chroot /sid-root /bin/bash` é fácil, mas isso mantém todos os tipos de variáveis de ambiente atribuídas que você pode não desejar, e tem outros problemas. Uma maneira muito melhor é executar outro processo de login em um terminal virtual separado onde você pode logar no chroot diretamente.

Como em sistemas Debian padrões, do `tty1` ao `tty6` são executados consoles Linux e no `tty7` é executado o Sistema X Window, vamos definir o `tty8` para um console chroot'ado como um exemplo. Depois de criar o sistema chroot como descrito em 'Executar um sabor diferente do Debian com `chroot`' on the previous page, digite a partir do shell root do sistema principal:

```
main # echo "8:23:respawn:/usr/sbin/chroot /sid-root "\
       "/sbin/getty 38400 tty8"  >> /etc/inittab
main # init q    # recarrega o init
```

Configurando o X para o `chroot`

Você quer executar o X e o GNOME mais novos seguramente em seu chroot? Isso é perfeitamente possível! O exemplo seguinte fará o GDM executar em um terminal virtual `vt9`.

Primeiro instale um sistema chroot usando o método descrito em 'Executar um sabor diferente do Debian com `chroot`' on the preceding page. A partir do root do sistema principal, copie os arquivos de configuração chaves para o sistema chroot.

```
main # cp /etc/X11/XF86Config-4 /sid-root/etc/X11/XF86Config-4
main # chroot /sid-root # ou use console chroot
chroot # cd /dev; /sbin/MAKEDEV generic ; cd -
chroot # apt-get install gdm gnome x-window-system
```

```
chroot # vi /etc/gdm/gdm.conf # faça s/vt7/vt9/ na seção [servers]
chroot # /etc/init.d/gdm start
```

Aqui, o arquivo `/etc/gdm/gdm.conf` foi editado para alterar o primeiro console virtual do `vt7` para o `vt9`.

Agora você pode alternar facilmente entre os ambientes X completos no sistema chroot e em seu sistema principal apenas alternando entre terminais virtuais Linux; por exemplo usando Ctrl-Alt-F7 e Ctrl-Alt-F9. Divirta-se!

[FIXME] Adicionar um comentário e ligar ao script init do ambiente gdm chroot'ado.

Executar outras distribuições com `chroot`

Um ambiente chroot de outra distribuição pode ser criado facilmente. Você instala um sistema em partições separadas usando o instalador da outra distribuição. Se a partição raiz dela estiver em */dev/hda9*.

```
main # cd / ; mkdir /outra-dist
main # mount -t ext3 /dev/hda9 /outra-dist
main # chroot /outra-dist /bin/bash
```

Então proceda como em 'Executar um sabor diferente do Debian com `chroot`' on page 131, 'Configurando o login para `chroot`' on the facing page, e 'Configurando o X para o `chroot`' on the preceding page.

Construir um pacote com o `chroot`

Existe um pacote chroot mais especializado, o `pbuilder`, que constrói um sistema chroot e compila um pacote dentro do chroot. Esse é um sistema ideal para ser usado para checar se as dependências de compilação de um pacote estão corretas, e para se certificar que dependências de compilação desnecessárias ou incorretas não existirão no pacote gerado.

8.6.35 Como checar ligações diretas (hard)

Você pode checar se dois arquivos são o mesmo arquivo com duas ligações diretas (hard links) usando :

```
$ ls -li arquivo1 arquivo2
```

8.6.36 Usar `mount` em um arquivo de uma imagem de disco rígido

Se *`arquivo.img`* contém uma imagem de um disco rígido e o disco rígido original possuía uma configuração de disco que tinha *xxxx* = (bytes/setor) * (setores/cilindro), então o comando a seguir irá montá-lo em `/mnt`:

```
# mount -o loop,offset=xxxx arquivo.img /mnt
```

Note que a maioria dos discos rígidos possui 512 bytes/setor.

8.6.37 Samba

O básico para obter arquivos do Windows:

```
# mount -t smbfs -o username=meunome,uid=my_uid,gid=my_gid \
        //servidor/compartilhamento /mnt/smb  # monta arquivos Windows para o Linu
# smbmount //servidor/compartilhamento /mnt/smb \
        -o "username=meunome,uid=my_uid,gid=my_gid"
# smbclient -L 192.168.1.2 # lista os compartilhamentos em um computador
```

Vizinhos Samba podem ser checados a partir do Linux usando:

```
# smbclient -N -L endereço_IP_de_seu_PC | less
# nmblookup -T "*"
```

8.6.38 Utilitários para sistemas de arquivos não nativos

Muitos sistemas de arquivo não nativos têm suporte no kernel Linux, então podemos acessá-los simplesmente montando dispositivos que contêm o sistema de arquivos. Para alguns sistemas de arquivos, há também algumas ferramentas especializadas para acessar os sistemas de arquivos sem montar os dispositivos. Isso é feito com programas de espaço de usuário, de forma que não é necessário suporte para o sistema de arquivos no kernel.

- `mtools`: para sistema de arquivo MSDOS (MS-DOS, Windows)
- `cpmtools`: para sistema de arquivo CP/M
- `hfsutils`: para sistema de arquivo HFS (Macintosh nativo)
- `hfsplus`: para sistema de arquivo HFS+ (Macintosh moderno)

Para criar e verificar sistema de arquivos MS-DOS FAT, o pacote `dosfstools` é útil.

8.7 Erros típicos a serem notados

Aqui há alguns exemplos de ações perigosas. Os impactos negativos serão aumentados se você estiver usando a conta privilegiada: `root`.

8.7.1 `rm -rf .*`

Em "`rm -rf .*`", "`.*`" expande para incluir "`.`" e "`..`", e se você tiver privilégios de escrita no diretório pai então você acabará removendo também todos os diretórios **próximos** ao seu diretório atual.

- "`rm -rf .`" : remove tudo sob o diretório atual e o próprio diretório atual.
- "`rm -rf *`" : remove todos os arquivos sem ponto e todos os diretórios sem ponto sob o diretório atual
- "`rm -rf .[^.]*`" : remove todos os arquivos com ponto e todos os diretórios com ponto sob o diretório atual.
- "`rm -rf .*`" : remove tudo sob o diretório pai e o próprio diretório pai.

8.7.2 `rm /etc/passwd`

A perda de alguns arquivos importantes como `/etc/passwd` por estupidez é ruim. O sistema Debian faz cópias de segurança deles regularmente em `/var/backups`. Quando você tiver que restaurar esses arquivos, você pode ter que ajustar as permissões apropriadas manualmente.

```
# cp /var/backups/passwd /etc/passwd
# chmod 644 /etc/passwd
```

Veja também 'Recuperação dos dados de seleção de pacotes' on page 90.

Capítulo 9

Ajustando um sistema Debian

Este capítulo descreve apenas o básico da configuração do sistema através da interface baseada em linha de comando. Antes de ler este capítulo, você deve ler 'Dicas para instalação do Sistema Debian' on page 27.

Se você está preocupado com a segurança então deve ler o Securing Debian Manual (`http://www.debian.org/doc/manuals/securing-debian-howto/`), que vem no pacote `harden-doc`.

9.1 Inicialização do sistema

O Debian usa o sistema de script de init System V. Veja 'O programa `init`' on page 21 para uma introdução.

9.1.1 Personalizando os scripts init

A maneira mais fácil de controlar o comportamento de um script init é alterando atribuições de variáveis de ambiente no arquivo nomeado como no script init no diretório `/etc/default`. [1] Por exemplo, `/etc/default/hotplug` pode ser usado controlar como `/etc/init.d/hotplug` funciona. O arquivo `/etc/default/rcS` pode ser usado para personalizar padrões de tempo de inicialização para `motd`, `sulogin`, etc.

Se você não puder obter o comportamento desejado mudando essas variáveis, então você pode modificar os próprios scripts init: eles são todos arquivos de configuração.

---

[1] Os arquivos em `/etc/default/` contêm **apenas** atribuições de variáveis de ambiente. Cada arquivo é usado como fonte pelo script init a ele correspondente de forma que essas atribuições cancelam quaisquer definições de variáveis no próprio script init. A escolha do nome do diretório é peculiar (`http://lists.debian.org/debian-devel/2003/debian-devel-200308/msg02114.html`) ao Debian. É, grosseiramente, o equivalente do diretório `/etc/sysconfig` encontrado no RedHat e outras distribuições.

9.1.2 Personalizando o log do sistema

O log do sistema pode ser configurado usando `/etc/syslog.conf`. Verifique o pacote `colorize` se quiser um programa para colorir os arquivos de log do sistema. Veja também `syslogd(8)` e `syslog.conf(5)`.

9.1.3 Otimizando o hardware

Há umas poucas configurações de otimização do hardware que o Debian deixa aos cuidados do administrador do sistema.

- `hdparm`
  - Otimização de acesso ao disco rígido. Muito eficiente.
  - Perigoso. Você precisa ler `hdparm(8)` primeiro.
  - `hdparm -tT` */dev/hda* para testar a velocidade de acesso ao disco.
  - `hdparm -q -c3 -d1 -u1 -m16` */dev/hda* para aumentar a velocidade de um moderno sistema IDE. (Isso pode ser perigoso.)
- `setcd`
  - Otimização de acesso a unidade de compact disc.
  - `setcd -x` *2* para diminuir para velocidade *2* x.
  - Consulte `setcd(1)`.
- `setserial`
  - Conjunto de ferramentas para gerenciamento de portas seriais.
- `scsitools`
  - Conjunto de ferramentas para gerenciamento de hardware SCSI.
- `memtest86`
  - Conjunto de ferramentas para gerenciamento de hardware de memória.
- `hwtools`
  - Conjunto de ferramentas para gerenciamento de hardware em baixo nível.
    - `irqtune`: muda a prioridade IRQ de dispositivos para permitir que dispositivos que precisem de alta prioridade e serviço rápido (por exemplo, portas seriais, modems) tenham a prioridade necessária. É possível aumentar a velocidade através de serial/modem em 3x.
    - `scanport`: varre o espaço de I/O a partir de 0x100 até 0x3ff procurando por dispositivos ISA instalados.
    - `inb`: um pequeno e rápido hack que lê uma porta de I/O e devolve o valor em hexadecimal e binário.
- `schedutils`
  - Utilidades de escalonamento do Linux.
  - Estão incluídos `taskset`, `irqset`, `lsrt`, e `rt`.

  - Junto com `nice` e `renice` (não incluídos), eles permitem total controle dos parâmetros de escalonamento de processos.

Montar um sistema de arquivos com opção `noatime` também é muito eficaz para aumentar a velocidade de acesso ao arquivo. Consulte `fstab(5)` e `mount(8)`.

Alguns hardwares podem ser ajustados diretamente pelo kernel Linux através do sistema de arquivos proc. Veja 'Ajustando o kernel pelo sistema de arquivos proc' on page 107.

Há muitos utilitários de configuração de hardware específicos. Muitos deles se destinam a necessidades específicas do PC laptop. Eis alguns pacotes interessantes disponíveis no Debian:

- `tpconfig` - Um programa para configurar dispositivos touchpad
- `apmd` - Utilitários para Gerenciamento Avançado de Energia (APM)
- `acpi` - Exibe informações sobre dispositivos ACPI
- `acpid` - Utilitários para usar ACPI
- `lphdisk` - prepara partição de hibernação para Phoenix NoteBIOS
- `sleepd` - põe um laptop para dormir durante inatividade
- `noflushd` - permite que discos rígidos ociosos diminuam o giro
- `big-cursor` - cursores de mouse maiores para o X
- `acme` - Habilita os "botões multimídia" encontrados em laptops
- `tpctl` - ferramentas de configuração de hardware do IBM ThinkPad
- `mwavem` - suporte a modem Mwave/ACP
- `toshset` - acessa muito da interface de hardware do laptop Toshiba
- `toshutils` - utilitários de laptop Toshiba
- `sjog` - um programa para usar o "Jog Dial" (roda de avanço) em laptops Sony Vaio
- `spicctrl` - programa controlador do Sony Vaio para ajustar o brilho da luz de fundo do LCD

Aqui, o ACPI é uma estrutura para o sistema de gerenciamento de energia mais nova que o APM.

Alguns desses pacotes necessitam de módulos especiais do kernel. Eles já são inclusos na última fonte do kernel em muitos casos. Se tiver problemas, você pode precisar aplicar o último patch para o kernel você mesmo.

9.2 Restringindo o acesso

9.2.1 Restringindo logins com PAM

PAM (Pluggable Authentication Modules) permite a você controlar como os usuários efetuam login.

```
/etc/pam.d/*           # arquivos de controle de PAM
/etc/pam.d/login       # arquivo de controle de PAM para login
/etc/security/*        # parâmetros de módulo PAM
/etc/securetty         # controla login de root pelo console (login)
/etc/login.defs        # controla as definições para login (login)
```

Modifique o conteúdo de `/etc/pam.d/login` como a seguir, se você quiser terminais de console inseguros mas sem senha sob seu próprio risco.

```
#auth       required   pam_unix.so nullok
auth       required   pam_permit.so
```

Pode-se aplicar truques similares para o `xdm`, `gdm`, ... , para console X sem senha.

Por outro lado, instale `cracklib2` e modifique `/etc/pam.d/passwd` como a seguir, se você desejar garantir uma boa segurança de senhas.

```
password required       pam_cracklib.so retry=3 minlen=6 difok=3
```

Uma senha de login para uma única vez, para ativação de conta, também pode ajudar. Para isso, use o comando passwd com a opção `-e` `passwd(1)`.

O número máximo de processos pode ser configurado com `ulimit -u 1000` em um shell Bash ou com definições em `/etc/security/limits.conf` de PAM. Outros parâmetros como `core` podem ser configurados de maneira similar. O valor inicial de `PATH` pode ser ajustado em `/etc/login.defs` antes do script de início do shell.

A documentação para PAM está no pacote `libpam-doc`. O *Guia dos Administradores de Sistema para Linux-PAM* cobre a configuração de PAM, quais módulos estão disponíveis, etc. A documentação também inclui *O Guia dos Desenvolvedores de Aplicação para Linux-PAM* e *O Guia dos Escritores de Módulos Linux-PAM*.

9.2.2 "Por que o GNU `su` não suporta o grupo `wheel`"

Essa é a famosa frase do final da velha página `info su` de Richard M. Stallman. Não se preocupe: o `su` atual no Debian usa PAM, então pode-se restringir a habilidade de usar o `su` para qualquer grupo usando `pam_wheel.so` em `/etc/pam.d/su`. O seguinte configurará o grupo `adm` em um sistema Debian como um equivalente do grupo `wheel` do BSD e permitirá `su` sem uma senha para seus membros.

```
# configuração anti-RMS em /etc/pam.d/su
auth       required   pam_wheel.so group=adm

# Membros Wheel capazes de usar su sem uma senha
auth       sufficient pam_wheel.so trust group=adm
```

9.2.3 Propósitos de grupos padrões

Alguns grupos interessantes:

- O grupo `root` é o grupo wheel padrão para o `su` se `pam_wheel.so` for usado sem o argumento `group=`.

- O grupo `adm` pode ler arquivos de log.
- O grupo `cdrom` pode ser usado localmente para dar acesso a uma unidade de CD-ROM a um conjunto de usuários.
- O grupo `floppy` pode ser usado localmente para dar acesso a uma unidade de disquetes a um conjunto de usuários.
- O grupo `audio` pode ser usado localmente para dar acesso a um dispositivo de audio a um conjunto de usuários.
- O grupo `src` é dono do código fonte, incluindo arquivos em `/usr/src`. Ele pode ser usado localmente para dar a um usuário a capacidade de gerenciar o código fonte do sistema.
- A participação no grupo `staff` é útil para administradores de sistema do tipo helpdesk ou junior, dando-lhes a capacidade de fazer coisas em `/usr/local` e criar diretórios em `/home`.

Para uma lista completa, veja a seção "FAQ" no Securing Debian Manual (`http://www.debian.org/doc/manuals/securing-debian-howto/`), que também pode ser encontrado como o pacote `harden-doc` no Woody. Além disso, o novo `base-passwd` (>3.4.6) contém uma lista oficial: `/usr/share/doc/base-passwd/users-and-groups.html`.

9.2.4 Trabalhando de forma mais segura – `sudo`

Meu uso do `sudo` é na maior parte uma proteção contra minha própria estupidez. Pessoalmente, eu acho que usar o `sudo` é uma alternativa melhor que sempre usar o sistema como root.

Instale o `sudo` e ative-o configurando as opções em `/etc/sudoers` (`http://www.debian.org/doc/manuals/debian-reference/examples/`). Verifique também o recurso do grupo `sudo` em `/usr/share/doc/sudo/OPTIONS`.

A configuração de exemplo disponibiliza aos membros do grupo "staff" o acesso a qualquer comando executado como root sob o `sudo` e também dá aos membros de "src" o acesso a executar comandos selecionados como root sob o `sudo`.

A vantagem do `sudo` é que ele requer apenas a senha de um usuário ordinário para efetuar o login, e a atividade é monitorada. Essa é uma bela maneira de dar alguma autoridade a um administrador junior. Por exemplo:

```
$ sudo chown -R myself:mygrp .
```

É claro que se você sabe a senha de root (como muitos usuários domésticos sabem), qualquer comando pode ser executado como root a partir de uma conta de usuário:

```
$ su -c "shutdown -h now"
Password:
```

(Eu sei que deveria diminuir os privilégios `sudo` das contas admin. Mas como esse é meu servidor caseiro, ainda não me preocupei como isso.)

Se quiser um programa diferente que permita que usuários ordinários executem comandos com privilégios de root, veja o pacote `super`.

9.2.5 Restringindo acesso a serviços

O *super-servidor* de Internet, `inetd`, é iniciado na inicialização por `/etc/rc2.d/S20inetd` (para RUNLEVEL=2), que é um link simbólico para `/etc/init.d/inetd`. Essencialmente, o `inetd` permite a execução de um daemon para a chamada de vários outros, reduzindo a carga no sistema.

Sempre que um pedido de um serviço chega, seu protocolo e serviço são identificados procurando-os nas bases de dados em `/etc/protocols` e `/etc/services`. O `inetd` então procura um serviço de Internet normal na base de dados de `/etc/inetd.conf` , ou um serviço baseado em Sun-RPC em `/etc/rpc.conf`.

Para segurança do sistema, não esqueça de desabilitar serviços sem uso em `/etc/inetd.conf`. Os serviços Sun-RPC precisam ser ativados para NFS e outros programas baseados em RPC.

Algumas vezes, o `inetd` não inicia um servidor desejado diretamente, mas inicia o programa daemon wrapper de TCP/IP `tcpd` com o nome do servidor desejado como seu argumento em `/etc/inetd.conf`. Nesse caso, o `tcpd` executa o programa servidor apropriado depois de registrar o pedido e fazer algumas verificações adicionais usando `/etc/hosts.deny` e `/etc/hosts.allow`.

Se você tiver problemas com acesso remoto em um sistema Debian recente, comente "ALL: PARANOID" em `/etc/hosts.deny` se ele existir.

Para mais detalhes, veja `inetd(8)`, `inetd.conf(5)`, `protocols(5)`, `services(5)`, `tcpd(8)`,`hosts_access(5)`,e `hosts_options(5)`.

Para mais informação sobre Sun-RPC, veja `rpcinfo(8)` ,`portmap(8)`,e `/usr/share/doc/portmap/portmapper.txt.gz`.

9.2.6 Centralizando a autenticação – LDAP

Use Lightweight Directory Access Protocol (LDAP)(Protocolo de Acesso a Diretório Leve) Referências:

- OpenLDAP (`http://www.openldap.org/`)
- Guia do Administrador OpenLDAP no pacote `openldap-guide`
- LDP: LDAP Linux HOWTO (`http://www.tldp.org/HOWTO/LDAP-HOWTO/index.html`)
- LDP: LDAP Implementation HOWTO (`http://www.tldp.org/HOWTO/LDAP-Implementation-HOWTO/index.html`)
- OpenLDAP, relatos de uso extensivo (`http://portal.aphroland.org/~aphro/ldap-docs/ldap.html`)
- Open LDAP com Courier IMAP e Postfix (`http://alinux.washcoll.edu/docs/plc/postfix-courier-howto.html`)

9.3 Gravadores de CD

Os gravadores de CD com interfaces ATAPI/IDE recentemente se tornaram uma opção muito popular. É uma boa mídia para backup do sistema e arquivamentos para o usuário doméstico que precise de capacidade < 640 MB. Para informação de maior autoridade, veja o CD-Writing-HOWTO (`http://www.tldp.org/HOWTO/CD-Writing-HOWTO.html`) do LDP.

9.3.1 Introdução

Primeiro, qualquer interrupção dos dados enviados para o gravador de CDs causará danos irrecuperáveis ao CD. Obtenha um gravador de CD com o maior buffer possível. Se dinheiro não for problema, não se aborreça com um ATAPI/IDE, simplesmente pegue uma versão SCSI. Se você tiver escolha sobre em qual interface IDE conectar, use a que estiver no barramento PCI (isto é, na placa mãe) ao invés de uma no barramento ISA (uma placa SB16, etc.).

Quando um gravador de CDs está conectado à IDE, precisa usar o driver IDE-SCSI ao invés de um driver de CD IDE ordinário para kernels Linux 2.2 e 2.4. Além disso, o driver SCSI genérico precisa ser ativado. Há duas possíveis maneiras de se fazer isso, considerando um kernel distribuído com distribuições atuais (como de março de 2001).

9.3.2 Alternativa 1: módulos + `lilo`

Se você estiver usando um kernel do Debian, adicione a seguinte linha ao `/etc/lilo.conf`. Se forem usadas várias opções, liste-as separadas por espaços:

```
append="hdx=ide-scsi ignore=hdx"
```

Aqui a localização do gravador de CDs, que é acessado através do driver ide-scsi, é indicada por `hdx`, onde *x* representa um dos seguintes:

```
hda          para o mestre na primeira porta IDE
hdb          para o escravo na primeira porta IDE
hdc          para o mestre na segunda porta IDE
hdd          para o escravo na segunda porta IDE
hde ... hdh  para uma unidade em uma porta IDE externa ou porta IDE ATA66/100
```

Digite os seguintes comandos como root para ativar depois de acabar toda a configuração:

```
# lilo
# shutdown -h now
```

9.3.3 Alternativa 2: recompilar o kernel

O Debian usa o `make-kpkg` para criar um kernel. Use o novo `--append_to_version` com o `make-kpkg` para criar várias imagens de kernel. Veja 'O kernel Linux no Debian' on page 101.

Use a seguinte configuração através do `make menuconfig`:

- bzImage
- Exclua o driver para CD IDE (não é necessário, mas isso simplifica)
- Compile ide-scsi e sg diretamente no kernel, ou como módulos

9.3.4 Passos posteriores à configuração

O suporte do kernel para o gravador de CDs pode ser ativado durante a inicialização com o seguinte:

```
# echo ide-scsi >>/etc/modules
# echo sg       >>/etc/modules
# cd /dev; ln -sf scd0 cdrom
```

A ativação manual pode ser feita com:

```
# modprobe ide-scsi
# modprobe sg
```

Depois de reinicializar, você pode verificar a instalação com:

```
$ dmesg|less
# apt-get install cdrecord
# cdrecord -scanbus
```

[Por Warren Dodge] Algumas vezes podem ocorrer conflitos entre `ide-scsi` e `ide-cd` se houver CD-ROM e CD-R/RW no sistema. Tente adicionar a seguinte linha ao seu `/etc/modutils/aliases`, executando `update-modules`, e reinicialize.

```
pre-install      ide-scsi      modprobe ide-cd
```

Isso faz com que o driver IDE seja carregado antes do `ide-scsi`. O driver IDE `ide-cd` toma o controle do ATAPI CD-ROM — qualquer coisa que não lhe tenha sido dito para **ignorar**. Isso deixa apenas os dispositivos ignorados para o `ide-scsi` controlar.

9.3.5 Arquivo imagem de CD (inicializável)

Para criar um CD-ROM de arquivos sob um `diretório-alvo/` como `cd-image.raw` (inicializável, formato Joliet TRANS.TBL habilitado; se não inicializável, retire as opções `-b` e `-c`), insira um disco de inicialização na unidade de disquetes e

```
# dd if=/dev/fd0 diretório-alvo/boot.img
# mkisofs -r -V volume_id -b boot.img -c bootcatalog -J -T \
        -o cd-image.raw diretório-alvo/
```

Um hack divertido é fazer um CD-ROM de DOS inicializável. Se um disquete de inicialização ordinário de DOS estiver no `boot.img` acima, o CD-ROM inicializará como se um disquete de DOS estivesse na primeira unidade de disquetes (A:). Fazer isso com freeDOS pode ser mais interessante.

Esse arquivo de imagem de CD pode ser inspecionado montando-o no dispositivo loop.

```
# mount -t iso9660 -o ro,loop cd-image.raw /cdrom
# cd /cdrom
# mc
# umount /cdrom
```

9.3.6 Gravar CDs (R, RW):

Primeiro teste com (considerando velocidade dupla)

```
# nice --10 cdrecord -dummy speed=2 dev=0,0 disk.img
```

Então se for tudo bem, grave no CD-R com

```
# nice --10 cdrecord -v -eject speed=2 dev=0,0 disk.img
```

Ou grave em um disco CD-RW com

```
# nice --10 cdrecord -v -eject blank=fast speed=2 dev=0,0 disk.img
```

Algumas unidades de CD-RW funcionam melhor com

```
# nice --10 cdrecord -v blank=all speed=2 dev=0,0 disk.img
```

seguido de

```
# nice --10 cdrecord -v -eject speed=2 dev=0,0 disk.img
```

São necessários esses dois passos para prevenir que timeouts de SCSI durante o apagamento interfiram no passo da gravação. O valor do argumento para o `nice` pode necessitar de alguns ajustes.

9.3.7 Fazer um arquivo imagem de um CD

Alguns CD-Rs e CDs comerciais têm setores ruins no fim que tornam impossível sua cópia através do `dd` (o CD do Windows 98 é um deles). O pacote `cdrecord` vem com o comando `readcd`. Use-o para copiar o conteúdo de qualquer CD para um arquivo imagem. Se for um disco de dados, monte-o e execute `df` para ver seu tamanho real. Divida o número mostrado em blocos (1 bloco = 1024 bytes) por 2 para obter o número de setores do CD real (1 setor = 2048 bytes). Execute o `readcd` com opções e use essa imagem de disco para gravar o CD-R/RW.

```
# readcd dev=target,lun,scsibusno # selecione função 11
```

Aqui, ajuste os 3 parâmetros para 0 para muitos casos. Geralmente o número de setores dados pelo `readcd` é excessivo! Use o número acima a partir de um mount real para melhores resultados.

Deve-se notar que o uso do `dd` tem alguns problemas se usado no CD-ROM. A primeira execução do comando `dd` pode causar uma mensagem de erro e pode resultar em uma imagem de disco mais curta com a perda do final. A segunda execução do comando `dd` pode resultar em uma imagem de disco maior com lixo anexo ao final em alguns sistemas se o tamanho do dado não for especificado. Somente a segunda execução do comando `dd` com o tamanho de dado correto especificado sem ejetar o CD depois de mensagem de erro parece evitar esses problemas. Se o tamanho da imagem mostrado pelo `df` for `46301184` blocos, use o seguinte comando duas vezes para obter a imagem correta (esta é minha informação empírica):

```
# dd if=/dev/cdrom of=cd.img bs=2048 count=$((46301184/2))
```

9.3.8 Imagens de CD do Debian

Para obter as últimas informações sobre os CDs do Debian CDs, visite a página do Debian CD (`http://www.debian.org/CD/`).

Se você tiver uma conexão rápida com a Internet, considere instalar pela rede usando:

- algumas imagens de disquete (`http://www.debian.org/distrib/floppyinst`).
- uma imagem de CD inicializável mínimo (`http://www.debian.org/CD/netinst/`).

Se você não tiver uma conexão rápida com a Internet, pense em comprar os CDs de vendedores de CD (`http://www.debian.org/CD/vendors/`).

Por favor, não jogue fora largura de banda baixando imagens de CD padrão a menos que seja um testador de imagem de CD (mesmo com o novo método jigdo).

Uma imagem de CD notável é o KNOPPIX - Live Linux Filesystem On CD (`http://www.knopper.net/knoppix/index-en.html`). Esse CD inicializa em um sistema Debian funcional sem instalar-se no disco rígido.

9.3.9 Fazer cópia de segurança do sistema em CD-R

Para arquivos de configuração chaves e arquivos de dados para o CD-R, use o script de backup de exemplo `backup(http://www.debian.org/doc/manuals/debian-reference/examples/)`. Veja também 'Copiar e arquivar um subdiretório inteiro' on page 114 e 'Backups diferenciais e sincronização de dados' on page 116.

9.3.10 Copiar um CD de músicas para CD-R

Não testado por mim:

```
# apt-get install cdrecord cdparanoia
# cdparanoia -s -B
# cdrecord dev=0,0,0 speed=2 -v -dao -eject defpregap=1 -audio *.wav
```

ou,

```
# apt-get install cdrdao #disk at once
# cdrdao read-cd --device /dev/cdrom --paranoia-mode 3 my_cd # lê o cd
# cdrdao write --device /dev/cdrom --speed 8 my_cd    # grava um CD novo
```

O `cdrdao` faz uma cópia real (sem intervalos, etc. . .).

9.4 X

O Sistema X Window é provido pelo XFree86 (`http://www.xfree86.org/`). Há duas versões principais do servidor X disponíveis no sistema Debian: XFree86 Versão 3.3 (XF3) e XFree86 Versão séries 4.x (XF4), ambas baseadas em especificações X11R6 pelo X.Org (`http://www.x.org/`).

Para informações básicas do X, refira-se a `X(7)`, ao XWindow-User-HOWTO (`http://www.tldp.org/HOWTO/XWindow-User-HOWTO.html`) do LDP, e ao Mini-HOWTO de Aplicações remotas no X (`http://www.tldp.org/HOWTO/mini/Remote-X-Apps.html`). Para um guia de usuário específico do Debian, leia o arquivo `/usr/share/doc/xfree86-common/FAQ.gz` fornecido no pacote `xfree86-common`. Esse contém uma revisão interessante e de autoridade sobre os problemas de mapeamento de teclado, feita por Branden Robinson.

'O servidor X' on page 149 um programa em um sistema local que mostra uma janela X e/ou desktop em um monitor do usuário (CRT, LCD) e aceita entradas de teclado e mouse.

'Clientes X' on page 151 um programa em um sistema (local ou remoto) que executa uma aplicação compatível com X.

Isso reverte o uso ordinário de "servidor" e "cliente" em outros contextos.

Há várias maneiras de se fazer o "servidor X" (lado do display) aceitar conexões remotas de um "cliente X" (lado da aplicação):

- Método `xhost`
  - o mecanismo de listagem de host (muito inseguro).
  - protocolo não encriptado (sujeito ao "eavesdropping attack").
  - Não use isso, se possível.
  - Veja 'Conectando a um servidor X remoto – `xhost`' on page 156 e `xhost(1x)`.
- Método `xauth`
  - o mecanismo de cookie mágico do MIT (inseguro, mas melhor que o `xhost`).
  - protocolo não encriptado (sujeito ao "eavesdropping attack").
  - use isso apenas para conexão local já que ele usa a CPU de forma menos intensa que `ssh -X`.
  - Veja 'Obtendo root no X' on page 158 e `xauth(1x)`.
- Métodos `xdm`, `wdm`, `gdm`, `kdm`, ...
  - o mecanismo de cookie mágico do MIT (inseguro como o `xauth`).
  - Veja `xdm(1x)` e `Xsecurity(7)` para informações básicas sobre o controle de acesso ao display do X.
  - Veja `wdm(1x)`, `gdm(8)`, e `kdm.options(5)` para mais informações, se esses estiverem instalados.
  - Veja '`init` System-V e níveis de execução (runlevels)' on page 99 para saber como desabilitar o `xdm` para ter um console Linux depois da inicialização sem ter que excluir o pacote `xdm`.
- Método **`ssh -X`**
  - mecanismo de redirecionamento através de shell seguro (**seguro**).
  - protocolo encriptado (um desperdício de recursos se usado localmente).
  - use isso para conexões remotas.
  - Veja 'Conectando a um servidor X remoto – `ssh`' on page 157.

Todos os métodos de conexão remota, exceto o `ssh`, requerem conexão TCP/IP habilitada no servidor X. Veja 'Usando o X sobre TCP/IP' on page 156.

9.4.1 Pacotes do X

Existem alguns (meta)pacotes disponibilizados para facilitar a instalação do sistema X no Woody.

`x-window-system-core` Este metapacote provê os componentes essenciais para uma estação de trabalho executando o Sistema X Window. Ele provê as bibliotecas X, um servidor X (`xserver-xfree86`), um conjunto de fontes, e um grupo de clientes X básicos e utilidades.

`x-window-system` Este metapacote provê praticamente todos os componentes do Sistema X Window desenvolvidos pelo Projeto XFree86, assim como um conjunto de programas acessórios historicamente populares. (Notavelmete, ele depende dos pacotes `x-window-system-core`, `twm`, e `xdm`, isto é, não é preciso instalar o `x-window-system-core` se você instalar esse.)

`xserver-common-v3` Arquivos e utilidades comuns aos servidores X XFree86 3.x (XF3)

`xserver-*` Pacotes de servidores XF3 adicionais para suportar hardware não suportado pelo novo servidor XF4 (`xserver-xfree86`) por qualquer razão que seja. Algumas antigas placas ATI mach64 não são suportadas no XF4, outras placas travam na versão XF4 do Woody, etc. (Para saber os pacotes disponíveis, use `apt-cache search xserver-|less`. Todos os servidores XF3 dependem do pacote `xserver-common-v3`.)

Em muitos casos, o pacote a instalar é o `x-window-system`. (Se você quiser login pelo console, desabilite o `xdm` como descrito em '"Deixe-me desabilitar o X na inicialização ¡" on page 111.)

9.4.2 Detecção de hardware para o X

Para habilitar detecção de hardware durante o estágio de configuração do X, instale os seguintes pacotes antes de instalar o sistema X:

- `discover` — sistema de identificação de hardware.
- `mdetect` — ferramenta de autodetecção do mouse.
- `read-edid` — ferramenta para obtenção de informação para monitores VESA PnP.

9.4.3 O servidor X

Veja `XFree86(1x)` para obter informações sobre o servidor X.

Para chamar o servidor X a partir de um console local:

```
$ startx -- :<display> vtXX
por exemplo:
$ startx -- :1 vt8 -bpp 16
... inicia no vt8 conectado a localhost:1 com modo 16 bpp
```

Os argumentos dados depois de `--` são para o servidor X.

Note que, ao usar um script `~/.xserverrc` para personalizar a inicialização do servidor X, certifique-se de fazer `exec` para o servidor X real. Se não fizer isso, o servidor X pode demorar a iniciar e sair. Por exemplo:

```
#!/bin/sh
exec /usr/bin/X11/X -dpi 100 -nolisten tcp
```

Configurando o servidor X (versão 4)

Para (re-)configurar um servidor XF4,

```
# dpkg-reconfigure --priority=low xserver-common
# dpkg-reconfigure --priority=low xserver-xfree86
```

gerará o arquivo `/etc/X11/XF86Config-4` e configurará o X usando o script `dexconf`.

Configurando o servidor X (versão 3)

Para (re-)configurar um servidor XF3, por exemplo, para ATI mach64,

```
# dpkg-reconfigure --priority=low xserver-common-v3
# dpkg-reconfigure --priority=low xserver-mach64
```

gerará o arquivo `/etc/X11/XF86Config` e configurará o X usando o script `xf86config-v3`.

Configurando o servidor X manualmente

No Woody, para adicionar personalizações do usuário ao arquivo `/etc/X11/XF86Config-4`, **não edite o arquivo de configuração entre o texto**:

```
### BEGIN DEBCONF SECTION
[snip]
### END DEBCONF SECTION
```

Ao invés disso, **adicione as personalizações antes do texto**. Por exemplo, para usar um dispositivo de vídeo personalizado, adicione alguma coisa modificando o seguinte texto no *início* do arquivo:

```
Section "Device"
  Identifier        "Custom Device"
  Driver            "ati"
  Option            "NoAccel"
EndSection
```

```
Section "Screen"
  Identifier  "Custom Screen"
  Device      "Custom Device"
  Monitor     "Generic Monitor"
  DefaultDepth 24
  Subsection "Display"
    Depth       8
    Modes       "1280x960" "1152x864" "1024x768" "800x600" "640x480"
  EndSubsection
  Subsection "Display"
    Depth       16
    Modes       "1280x960" "1152x864" "1024x768" "800x600" "640x480"
  EndSubsection
  Subsection "Display"
    Depth       24
    Modes       "1280x960" "1152x864" "1024x768" "800x600" "640x480"
  EndSubsection
EndSection

Section "ServerLayout"
    Identifier       "Custom"
    Screen           "Custom Screen"
    InputDevice      "Generic Keyboard" "CoreKeyboard"
    InputDevice      "Configured Mouse" "CorePointer"
EndSection
```

No Sarge (`testing` no momento em que escrevo), se você quiser manter as personalizações do usuário no arquivo `/etc/X11/XF86Config` ao atualizar, execute os seguintes comandos como root:

```
# cp /etc/X11/XF86Config-4 /etc/X11/XF86Config-4.custom
# md5sum /etc/X11/XF86Config-4 > /var/lib/xfree86/XF86Config-4.md5sum
# dpkg-reconfigure xserver-xfree86
```

Para obter **fontes menos feias**, você precisa editar o `/etc/X11/XF86Config-4` como descrito em 'Fontes TrueType no X' on page 160.

Por favor, verifique também as outras partes de sua configuração do X. Ajustes ruins de monitor podem ser uma dor de cabeça até pior que fontes ruins, então certifique-se de que sua taxa de atualização é tão alta quanto seu monitor pode suportar (85 Hz é ótimo, 75 Hz está bom, 60 Hz é horrível.).

9.4.4 Clientes X

Muitos programas clientes X podem ser iniciados com um comando como esse:

```
cliente $ xterm -geometry 80x24+30+200 -fn 6x10 -display nomemáquina:0 &
```

Aqui, os argumentos opcionais da linha de comando significam:

- `-geometry` *`LARGURA`*`x`*`ALTURA`*`+`*`DESL_X`*`+`*`DESL_Y`*: o tamanho inicial e a posição da janela.
- `-fn` *`NOMEFONTE`*: a fonte a usar para exibir texto. *`NOMEFONTE`* pode ser:
  - a14: Fonte de tamanho normal
  - a24: Fonte de tamanho grande
  - ... (verifique as fontes disponíveis com `xlsfont`.)
- `-display` *`nomedisplay`*: o nome do servidor X a usar. *`nomedisplay`* pode ser:
  - *`nomemáquina`*`:`*`D.T`* significa tela *`T`* no display *`D`* da máquina *`nomemáquina`*; o servidor X para esse display está escutando na porta TCP 6000+D.
  - *`máquina`*`/unix:`*`D.T`* significa tela *`T`* no display *D* da máquina *`máquina`*; o servidor X para esse display está escutando o UNIX domain socket `/tmp/.X11-unix/XD` (então ele só é acessível pela *`máquina`*).
  - *`:D.T`* é equivalente ao *`máquina`*`/unix:`*`D.T`*, onde *`máquina`* é o nome da máquina local.

O *`nomedisplay`* padrão para o programa cliente X (lado da aplicação) pode ser definido pela variável de ambiente DISPLAY. Por exemplo, antes de executar um programa cliente X, a execução de um dos seguintes comandos faz isso:

```
$ export DISPLAY=:0
        # O padrão, máquina local usando a primeira tela do X
$ export DISPLAY=nomemaquina.dominio.nome:0.2
$ export DISPLAY=localhost:0
```

Sua inicialização pode ser personalizada em `~/.xinitrc`. Por exemplo:

```
xrdb -load $HOME/.Xresources
xsetroot -solid gray &
xclock -g 50x50-0+0 -bw 0 &
xload -g 50x50-50+0 -bw 0 &
xterm -g 80x24+0+0 &
xterm -g 80x24+0-0 &
twm
```

Como descrito em 'Sessões X personalizadas' on the facing page, isso substitui tudo que a execução normal de Xsession faz ao iniciar a partir de `startx`. Use o `~/.xsession` ao invés disso e use essa alternativa apenas como último recurso. Veja `xsetroot(1x)`, `xset(1x)` e 'Recursos do X' on page 157.

9.4.5 Sessões X

Uma sessão X (servidor X + cliente X) pode ser iniciada por:

- `startx`: comando de script para `xinit` para iniciar o servidor e o cliente X a partir do console de caracteres do Linux. Se o arquivo `~/.xinitrc` não existir, `/etc/X11/Xsession` é executado através de `/etc/X11/xinit/xinitrc`.
- `xdm`, `gdm`, `kdm`, ou `wdm`: Daemons gerenciadores de display X para iniciar o servidor e o cliente X, e para controlar o login a partir de uma tela GUI. `/etc/X11/Xsession` é executado diretamente.

O console pode ser disponibilizado como em '"Deixe-me desabilitar o X na inicialização ¡" on page 111.

Sessões X personalizadas

O script de inicialização padrão `/etc/X11/Xsession` é efetivamente uma combinação de `/etc/X11/Xsession.d/50xfree86-common_determine-startup` e `/etc/X11/Xsession.d/99xfree86-common_start`.

A execução de `/etc/X11/Xsession` é um pouco afetada por `/etc/X11/Xsession.options` e é essencialmente uma execução de um programa que foi encontrado primeiro na seguinte ordem com o comando `exec` :

1 `~/.xsession` ou `~/.Xsession`, se for definido.
2 `/usr/bin/x-session-manager`, se for definido.
3 `/usr/bin/x-window-manager`, se for definido.
4 `/usr/bin/x-terminal-emulator`, se for definido.

O significado exato desses comandos é determinado pelo sistema de alternativas do Debian descrito em 'Comandos alternativos' on page 98. Por exemplo:

```
# update-alternatives --config x-session-manager
... ou
# update-alternatives --config x-window-manager
```

Para tornar qualquer gerenciador de janelas do X como padrão enquanto se mantém os gerenciadores de sessão do GNOME e KDE instalados, substitua o `/etc/X11/Xsession.d/50xfree86-common_determine-startup` com o anexo no segundo relatório de bug em `http://bugs.debian.org/168347` (Eu espero que ele seja incluído logo.) e edite o `/etc/X11/Xsession.options` como a seguir para desabilitar o gerenciador de sessão X:

```
# /etc/X11/Xsession.options
#
# opções de configuração para /etc/X11/Xsession
# Veja Xsession.options(5) para uma explicação das opções disponíveis.
# Padrão habilitado
allow-failsafe
```

```
allow-user-resources
allow-user-xsession
use-ssh-agent
# Padrão desabilitado (habilite-os descomentando)
do-not-use-x-session-manager
#do-not-use-x-window-manager
```

Sem a modificação no sistema mencionada acima, `gnome-session` e `kdebase` são os pacotes contendo esses gerenciadores de sessão X. Sua remoção permite que o gerenciador de janelas X seja um padrão. (Yack, alguma idéia melhor?)

Em um sistema onde `/etc/X11/Xsession.options` contém uma linha `allow-user-xsession` sem caracteres precedendo, qualquer usuário que defina um `~/.xsession` ou `~/.Xsession` poderá personalizar a ação de `/etc/X11/Xsession`.

O último comando no arquivo `~/.xsession` deve usar a forma `exec` *`algum-gerenciador-de-janelas/sessão`* para iniciar seu gerenciador de janelas/sessão X.

Um bom exemplo de um script `~/.xsession` é dado em `/usr/share/doc/xfree86-common/examples/xsession.gz`.

Eu uso isso para definir o gerenciador de janelas acesso à tela e idioma de suporte para cada conta de usuário. Veja 'Iniciando uma sessão X para um usuário' on this page, 'Obtendo root no X' on page 158, e 'Exemplo para um sistema X window multi-idiomas' on page 176.

Se você desejar ter vários programas clientes X iniciados automaticamente, veja os exemplos 'Clientes X' on page 151 e invoque-os a partir do arquivo `~/.xsession` ao invés do arquivo `~/.xinitrc`.

Recursos X específicos do usuário podem ser definidos em `~/.Xresources`. Veja 'Recursos do X' on page 157.

Mapas de teclado e funções dos botões do apontador personalizados para o usuário no X também podem ser especificados no script de inicialização do usuário. Veja 'Mapas de teclado e funções de botões do apontador no X' on page 158.

Iniciando uma sessão X para um usuário

Seguindo o princípio descrito em 'Sessões X personalizadas' on the preceding page, um gerenciador de sessão/janelas X específico do usuário pode ser ativado instalando o pacote indicado e definindo o conteúdo no final do arquivo `~/.xsession` como a seguir. (Eu gosto do `blackbox/fluxbox` por seu estilo simples e pela alta velocidade.):

- gerenciador de sessão X padrão
  - Veja 'Comandos alternativos' on page 98
  - `exec /usr/bin/x-session-manager`
- gerenciador de janelas X padrão

  - Veja ‘Comandos alternativos’ on page 98
  - `exec /usr/bin/x-window-manager`

- Gerenciador de sessão do GNOME (pesado)
  - Instalar pacote: `gnome-session`
  - `exec /usr/bin/gnome-session`

- Gerenciador de sessão do KDE (pesado)
  - Instalar pacote: `kdebase` (ou `kdebase3` para KDE3)
  - `exec /usr/bin/kde2`

- Gerenciador de janelas Blackbox (leve, simples)
  - Instalar pacote: `blackbox`
  - `exec /usr/bin/blackbox`

- Gerenciador de janelas Fluxbox (leve, novo blackbox)
  - Instalar pacote: `fluxbox`
  - `exec /usr/bin/fluxbox`

- Gerenciador de janelas Xfce (parecido com Mac OS-X, SUN CDE)
  - Instalar pacote: `xfce`
  - `exec /usr/bin/xfwm`

- Gerenciador de janelas IceWM (leve, alternativa ao GNOME)
  - Instalar pacote: `icewm`
  - `exec /usr/bin/X11/icewm`

- Gerenciador de janelas virtual FVWM2 (leve, parecido com Win95)
  - Instalar pacote: `fvwm`
  - `exec /usr/bin/fvwm2`

- Gerenciador de janelas Windowmaker (algo parecido com NeXT)
  - Instalar pacote: `wmaker`
  - `exec /usr/bin/wmaker`

- Gerenciador de janelas Enlightenment (pesado).
  - Instalar pacote: `enlightenment`
  - `exec /usr/bin/enlightenment`

Veja Gerenciadores de Janela para o X (`http://www.xwinman.org`).

Configurando KDE e GNOME

Para configurar um ambiente KDE ou GNOME completo, os seguintes metapacotes são úteis:
- KDE: instale o pacote `kde`
- GNOME: instale o pacote `gnome`

A instalação desses pacotes com ferramentas que manipulam `recommends`, como o `dselect` e o `aptitude`, lhe dá melhores opções de software que simplesmente instalá-los com o `apt-get`.

Se você quiser login de console, certifique-se de desabilitar gerenciadores de sessão como `kdm`, `gdm`, e `wdm`, que podem ser inseridos pelas dependências, como descrito em '"Deixe-me desabilitar o X na inicialização ¡" on page 111.

Se você quiser ter o GNOME como padrão do sistema sobre o KDE, não esqueça de configurar `x-session-manager` como em 'Comandos alternativos' on page 98.

9.4.6 Usando o X sobre TCP/IP

Devido a conexão com soquete TCP/IP remota sem encriptação ser sujeita a um "eavesdropping attack", a configuração padrão para o X em versões recentes do Debian desabilita o soquete TCP/IP. Pense em usar o `ssh` para uma conexão remota do X (veja 'Conectando a um servidor X remoto – `ssh`' on the next page).

O método descrito aqui não é encorajado a menos que se esteja em um ambiente muito seguro atrás de um bom sistema de firewall com apenas usuários conhecidos presentes. Use o seguinte comando para verificar a configuração atual de seu servidor X para o soquete TCP/IP:

```
# find /etc/X11 -type f -print0 | xargs -0 grep nolisten
/etc/X11/xinit/xserverrc:exec /usr/bin/X11/X -dpi 100 -nolisten tcp
```

Remova `-nolisten` para restaurar escuta TCP/IP no servidor X.

9.4.7 Conectando a um servidor X remoto – `xhost`

O `xhost` permite acesso baseado em nomes de máquinas. Isso é muito inseguro. O seguinte desabilitará a verificação de máquina e permitirá conexões de qualquer lugar se uma conexão com soquete TCP/IP for permitida (veja 'Usando o X sobre TCP/IP' on this page):

```
$ xhost +
```

Você pode reabilitar a verificação de máquina com:

```
$ xhost -
```

O `xhost` não distingue entre diferentes usuários na máquina remota. Além disso, nomes de máquinas (endereços) podem ser falsificados.

Esses método precisa ser evitado mesmo com o critério de máquina mais restritivo se você estiver em uma rede não confiável (por exemplo com uma conexão de acesso discado PPP à Internet). Veja `xhost(1x)`.

9.4.8 Conectando a um servidor X remoto – `ssh`

O uso de `ssh` permite uma conexão segura a partir de um servidor X local para um servidor de aplicações remoto.

- Defina as entradas `X11Forwarding` e `AllowTcpForwarding` para `yes` no `/etc/ssh/sshd_config` da máquina remota, se você quiser evitar as opções de linha de comando correspondentes.
- Inicie o servidor X na máquina local.
- Abra um `xterm` na máquina local.
- Execute o `ssh` para estabelecer uma conexão com o sistema remoto.

```
nomelocal @ maquinalocal $ ssh -q -X -l nomelogin maquinaremota.dominio
Password:
.....
```

- Execute os comandos da aplicação X no sistema remoto.

```
nomelogin @ maquinaremota $ gimp &
```

Esse método permite a exibição da saída de um cliente X remoto como se ele estivesse conectado localmente através de um soquete de domínio UNIX local.

9.4.9 O emulador de terminal X – `xterm`

Aprenda tudo sobre o `xterm` em `http://dickey.his.com/xterm/xterm.faq.html`.

9.4.10 Recursos do X

Muitos programas X antigos, como o `xterm`, usam a base de dados de recursos do X para configurar sua aparência. O arquivo `~/.Xresources` é usado para armazenar as especificações dos recursos do usuário. Esse arquivo é carregado automaticamente nos recursos padrões do X ao fazer o login. Os padrões gerais do sistema para os recursos do X são armazenados em `/etc/X11/Xresources/*` e os padrões de aplicações deles são armazenados em `/etc/X11/app-defaults/*`. Use esses ajustes como pontos de partida.

Aqui estão algumas configurações úteis para adicionar ao seu arquivo `~/.Xresources`:

```
! Define a fonte para uma mais legível 9x15
XTerm*font: 9x15
```

```
! Exibe uma barra de rolagem
XTerm*scrollBar: true

! Define o tamanho do buffer para 1000 linhas
XTerm*saveLines: 1000

! Tela grande do kterm
KTerm*VT100*fontList: -*-fixed-medium-r-normal--24-*,\
 -*-gothic-medium-r-normal--24-*,\
 -*-mincho-medium-r-normal--24-*
```

Para fazer com que essas configurações tenham efeito imediato, carregue-as na base de dados com o comando:

```
xrdb -merge ~/.Xresources
```

Veja `xrdb(1x)`.

9.4.11 Mapas de teclado e funções de botões do apontador no X

O programa `xmodmap` é usado para editar e exibir o mapa modificador do teclado e a tabela de mapa do teclado que são usados pelas aplicações clientes para converter códigos de tecla de eventos em símbolos de tecla no X.

```
$ xmodmap -pm
 ... exibe o mapa modificador atual
$ xmodmap -pk | pager
 ... exibe a tabela do mapa atual
$ xmodmap -e "pointer = 3 2 1" # define mouse para a mão esquerda
$ xmodmap ~/.xmodmaprc # configura o teclado como em ~/.xmodmaprc
```

Ele geralmente é executado a partir do script de inicialização de sessão do usuário, `~/.xsession`.

Para obter o `código de tecla` (keycode), execute `xev` no X e pressione teclas. Para obter o significado do `símbolo de tecla` (keysym), procure na definição MACRO no arquivo `/usr/include/X11/keysymdef.h`. Todas as expressões `#define` nesse arquivo são nomeadas como `XK_` antecedendo os nomes de `símbolo de tecla`.

Veja `xmodmap(1x)`.

9.4.12 Obtendo root no X

Se um programa GUI precisar ser executado com privilégios de root, use os seguintes procedimentos para exibir a saída do programa em um servidor X do usuário. **Nunca inicie um servidor X diretamente da conta root** para evitar possíveis riscos de segurança.

Inicie o servidor X como um usuário normal e abra um console `xterm`. Então:

```
$ XAUTHORITY=$HOME/.Xauthority
$ export XAUTHORITY
$ su root
Password:*****
# printtool &
```

Ao usar esse truque para `su` para um usuário não root, assegure-se de que `~/.Xauthority` pode ser lido por grupo para esse usuário não root.

Para automatizar essa seqüência de comandos, crie um arquivo `~/.xsession` a partir da conta do usuário, contendo as seguintes linhas:

```
# Isso faz o X funcionar quando uso su para conta root.
if [ -z "$XAUTHORITY" ]; then
        XAUTHORITY=$HOME/.Xauthority
        export XAUTHORITY
fi
unset XSTARTUP
# Se for desejado um gerenciador de janelas/sessão específico, descomente
# a linha seguinte e edite-a para preencher suas necessidades.
#XSTARTUP=/usr/bin/blackbox
# Isso inicia programa gerenciador de sessão/janela x
if [ -z "$XSTARTUP" ]; then
  if [ -x /usr/bin/x-session-manager ]; then
    XSTARTUP=x-session-manager
  elif [ -x /usr/bin/x-window-manager ]; then
    XSTARTUP=x-window-manager
  elif [ -x /usr/bin/x-terminal-emulator ]; then
    XSTARTUP=x-terminal-emulator
  fi
fi
# executa gerenciador de janela/sessão X auto selecionado
exec $XSTARTUP
```

Então execute `su` (não `su -`) em uma janela `xterm` do usuário. Agora programas GUI iniciados a partir desse `xterm` podem exibir saída na janela desse usuário enquanto é executado com privilégio de root. Esse truque funciona sempre que o `/etc/X11/Xsession` padrão é executado. Se um usuário definir sua personalização usando `~/.xinit` ou `~/.xsession`, a variável de ambiente `XAUTHORITY` acima mencionada também precisa ser definida similarmente naqueles scripts.

Alternativamente, o `sudo` pode ser usado para automatizar a seqüência de comando:

```
$ sudo xterm
```

```
... ou
$ sudo -H -s
```

Aqui `/root/.bashrc` deve conter:

```
if [ $SUDO_USER ]; then
    sudo -H -u $SUDO_USER xauth extract - $DISPLAY | xauth merge -
fi
```

Isso funciona bem mesmo com o diretório home do usuário em uma montagem NFS, porque root não lê o arquivo `.Xauthority`.

Há também vários pacotes especializados para esse propósito: `kdesu`, `gksu`, `gksudo`, `gnome-sudo`, e `xsu`. Alguns outros métodos podem ser usados para se obter resultados similares: criar um link simbólico a partir de `/root/.Xauthority` para o do usuário correspondente; uso do script sux (`http://fgouget.free.fr/sux/sux-readme.shtml`); ou colocar ”`xauth merge ~`*`USER_RUNNING_X`*`/.Xauthority`“ no script de inicialização root.

Veja mais na lista de mensagens debian-devel (`http://lists.debian.org/debian-devel/2002/debian-devel-200207/msg00259.html`).

9.4.13 Fontes TrueType no X

O `xfs` padrão do XFree86-4 funciona bem com fontes TrueType. Se você estiver usando o XFree86-3, precisa instalar um servidor de fontes de terceiros como o `xfs-xtt`.

Você só precisa ter certeza de que as aplicações com as quais quer usar as fontes TrueType foram ligadas usando libXft ou libfreetype (provavelmente você nem precisa preocupar-se com isso se estiver usando .debs pré-compilados).

Primeiro configure a estrutura de suporte a fontes:

- Instale os pacotes `x-ttcidfont-conf` e `defoma`. Isso automatiza a geração de arquivos `fonts.scale` e `fonts.dir`.
  ```
  # apt-get install x-ttcidfont-conf
  ```
- Edite o arquivo `/etc/X11/XF86Config-4` em `Section ”Files“` assim:
  ```
  Section "Files"
      FontPath  "/var/lib/defoma/x-ttcidfont-conf.d/dirs/TrueType"
      FontPath  "/usr/share/fonts/truetype"
      FontPath  "/usr/lib/X11/fonts/CID"
      FontPath  "/usr/lib/X11/fonts/Speedo"
      FontPath  "/usr/lib/X11/fonts/misc"
      FontPath  "/usr/lib/X11/fonts/cyrillic"
      FontPath  "/usr/lib/X11/fonts/100dpi:unscaled"
      FontPath  "/usr/lib/X11/fonts/75dpi:unscaled"
      FontPath  "/usr/lib/X11/fonts/Type1"
  EndSection
  ```

> A primeira linha ajustará o XFree86 para usar quaisquer fontes TrueType que você instalar a partir de pacotes Debian. A entrada de fonte Type1 é movida para baixo já que o XFree86 faz um trabalho um tanto podbre de renderização de fontes Type1. O truque de `:unscaled` para fontes bitmap não deve mais ser necessário para o novo XF4 mas eu incluí aqui só por segurança. Para preservar as alterações manuais feitas no arquivo `/etc/X11/XF86Config-4`, siga as instruções em 'Configurando o servidor X manualmente' on page 150.

Então instale pacotes de fontes DFSG:

- Fontes TrueType ocidentais:
  - `ttf-bitstream-vera`: Um conjunto de fontes TrueType de alta qualidade criadas pela Bitstream, Inc. [2]
  - `ttf-freefont`: Um conjunto de fontes TrueType livres de alta qualidade cobrindo o conjunto de caracteres UCS.
  - `ttf-thryomanes`: Uma fonte TrueType Unicode cobrindo Latin, Grego, Cirílico e IPA.
- Fontes asiáticas:
  - `tfm-arphic-bsmi00lp`: Fonte TrueType Arphic Chinês "AR PL Mingti2L Big5" dado métrico fonte TeX
  - `tfm-arphic-bkai00mp`: Fonte TrueType Arphic Chinês "AR PL KaitiM Big5" dado métrico fonte TeX
  - `tfm-arphic-gbsn00lp`: Fonte TrueType Arphic Chinês "AR PL SungtiL GB" dado métrico fonte TeX
  - `tfm-arphic-gkai00mp`: Fonte TrueType Arphic Chinês "AR PL KaitiM GB" dado métrico fonte TeX
  - `ttf-baekmuk`: Séries de fontes TrueType Baekmuk Coreano
  - `hbf-jfs56`: Fonte bitmap Jianti Fangsong 56x56 Chinês (GB2312) para CJK
  - `hbf-cns40-b5`: Fonte bitmap Fanti Song 40x40 Chinês (Big5) para CJK
  - `hbf-kanji48`: Fonte bitmap Kanji 48x48 Japonês (JIS X-0208) para CJK

Já que fontes **Livres** são algumas vezes limitadas, a instalação ou compartilhamento de algumas fontes comerciais TrueType é uma opção para usuários Debian. Para facilitar esse processo para o usuário, alguns pacotes de conveniência foram criados:

- `ttf-commercial`
- `msttcorefonts (>1.1.0)` [3]

Você terá uma seleção realmente boa de fontes TrueType sob o custo de contaminar seu sistema **Livre** com fontes não-Livres.

Todas esses pacotes de fontes no Debian devem funcionar sem nenhum esforço e aparecer disponíveis para todos os programas X que usam o sistema de fontes "principal" regular. Isso inclui coisas como Xterm, Emacs e muitas outras aplicações não-KDE e não-Gnome.

Agora, execute o `xfontsel` e selecione quaisquer fontes TrueType no menu fndry, você deve ser capaz de ver muitas entradas não acinzentadas no menu "fmly".

---

[2] Apesar disso não estar disponível no Woody, você pode instalar a partir do Sarge.

[3] O pacote do Woody não funciona desde 8/2002 devido a mudanças na página web da Microsoft. Ao invés disso, use a versão do Sarge mesmo no Woody.

Para o KDE2.2 e GNOME1.4 (com libgdkxft0, que é um hack para fazer o GTK 1.2 renderizar fontes com anti-alias), você precisa configurar Xft1, também. Xft1 está altamente desatualizado, e basicamente só é usado pelo GNOME1.4 e KDE2.2. Edite o arquivo `/etc/X11/XftConfig` e adicione uma linha como

```
        dir "/var/lib/defoma/x-ttcidfont-conf.d/dirs/TrueType"
```

antes das outras linhas dir. [4] Para o GNOME2 e KDE3 (pós edição do Sarge), você precisa configurar o `fontconfig` que o Xft2 usa para encontrar fontes. [5] Você não deve precisar instalar nada extra para isso, já que todos os pacotes que usam `fontconfig` já irão Depender dele (indiretamente).

Primeiro, veja em `/etc/fonts/fonts.conf`. Deve haver uma linha como a abaixo. Se não houver, abra o `/etc/fonts/local.conf` e adicione essa

```
        <dir>/var/lib/defoma/x-ttcidfont-conf.d/dirs/TrueType</dir>
```

logo abaixo da linha <fontconfig>.

O `Fontconfig` deve pegá-las imediatamente, e "fc-list" deve listar suas novas fontes. Outro recurso elegante do `fontconfig` é que você pode simplesmente colocar fontes em `~./fonts/` e todos os seus programas que usam `fontconfig` terão acesso a elas imediatamente.

Se você instalar manualmente um novo conjunto de fontes TrueType enquanto estiver no X sem usar pacote Debian, execute

```
   # xset fp rehash
```

para fazer com que o XFree86 olhe os conteúdos daquele diretório novamente e pegue as novas.

9.4.14 Navegadores Web no X

Há alguns pacotes de navegadores Web com capacidades de exibição gráfica desde a versão Woody:

- `mozilla` O navegador Mozilla (novo)
- `galeon` Navegador baseado no Mozilla com uma interface de usuário para o Gnome (novo)
- `konqueror` Navegador KDE
- `dillo` Navegador GTK
- `amaya-gtk` Navegador de referência W3C

---

[4] Eu não tenho mais nada de xft1 em minha máquina, então não tenho certeza se você precisa reiniciar o X ou não antes de que isso tenha efeito. Pelo que me lembro "xftcache" atualizaria o cache de Xft1, mas seria bom se alguém pudesse confirmar isso para mim.

[5] `Fontconfig` **não** existe no Woody.

- `amaya-lesstif` Navegador de referência W3C
- `netscape-...` (muito, antigo)
- `communicator-...` (muito, antigo)
- ...

A versão do `mozilla` precisa combinar com a versão que o `galeon` requer. Apesar deles diferirem na interface para o usuário, esses dois programas compartilham o mecanismo de processamento HTML Gecko.

Os plug-ins para navegadores como `mozilla` e `galeon` podem ser habilitados instalando os "`*.so`" manualmente no diretório de plug-in e reiniciando os navegadores.

Recursos plug-in:
- Plug-in Java: instale o binário "J2SE" de `http://java.sun.com`.
- Flash plug-in: instale o binário "Macromedia Flash Player 5" de `http://www.macromedia.com/software/flashplayer/`.
- `freewrl`: navegador VRML e plug-in Netscape
- ...

9.5 SSH

O SSH (Secure SHell) é a maneira segura de conectar através da Internet. Uma versão livre de SSH chamada OpenSSH está disponível como o pacote `ssh` no Debian.

9.5.1 Básico do SSH

Primeiro instale o servidor e o cliente OpenSSH.

```
# apt-get update && apt-get install ssh
```

O arquivo `/etc/ssh/sshd_not_to_be_run` não pode estar presente se desejar-se executar o servidor OpenSSH.

O SSH tem dois protocolos de autenticação:

- protocolo SSH versão 1:
  - a versão Potato suporta apenas esse protocolo.
  - métodos de autenticação disponíveis:
    - RSAAuthentication: autenticação de usuário baseado em chave de identidade RSA
    - RhostsAuthentication: autenticação de máquina baseado em .rhosts (inseguro, desabilitado)
    - RhostsRSAAuthentication: autenticação .rhosts combinada com chave de máquina RSA (desabilitado)
    - ChallengeResponseAuthentication: autenticação pergunta-resposta RSA

    - PasswordAuthentication: autenticação baseada em senha

- protocolo SSH versão 2:
  - versões pós-Woody usam esse como o protocolo primário.
  - métodos de autenticação disponíveis:
    - PubkeyAuthentication: autenticação de usuário baseado em chave pública
    - HostbasedAuthentication: autenticação `.rhosts` ou `/etc/hosts.equiv` combinado com autenticação de chave pública de máquina cliente (desabilitado)
    - ChallengeResponseAuthentication: autenticação pergunta-resposta
    - PasswordAuthentication: autenticação baseada em senha

Tenha cuidado com essas diferenças se você estiver migrando para o Woody ou estiver usando um sistema não-Debian.

Veja `/usr/share/doc/ssh/README.Debian.gz`, `ssh(1)`, `sshd(8)`, `ssh-agent(1)`, e `ssh-keygen(1)` para detalhes.

Os arquivos de configuração chaves são os seguintes:

- `/etc/ssh/ssh_config`: padrões do cliente SSH. Veja `ssh(1)`. As entradas notáveis são:
  - `Host`: Limita as declarações seguintes (até a próxima palavra-chave Host) para serem apenas para aquelas máquinas que combinam com um dos padrões dados depois da palavra-chave.
  - `Protocol`: Especifica as versões de protocolo SSH. O padrão é "2,1".
  - `PreferredAuthentications`: Especifica o método de autenticação do cliente SSH2. O padrão é "hostbased,publickey,keyboard-interactive,password".
  - `PasswordAuthentication`: Se você quiser efetuar login com uma senha, precisa certificar-se de que isso não está ajustado para `no`.
  - `ForwardX11`: O padrão é desabilitado. Isso pode ser substituído pela opção de linha de comando "`-X`".

- `/etc/ssh/sshd_config`: Padrões do servidor SSH. Veja `sshd(8)`. As entradas notáveis são:
  - `ListenAddress`: Especifica os endereços locais em que o `sshd` deve escutar. São permitidas múltiplas opções.
  - `AllowTcpForwarding`: O padrão é desabilitado.
  - `X11Forwarding`: O padrão é desabilitado.

- `$HOME/.ssh/authorized_keys`: as listas das chaves públicas padrões que os clientes usam para conectar a essa conta nessa máquina. Veja `ssh-keygen(1)`.

- `$HOME/.ssh/identity`: Veja `ssh-add(1)` e `ssh-agent(1)`.

O seguinte iniciará uma conexão `ssh` a partir de um cliente.

```
$ ssh nomeusuario@nomemaquina.dominio.ext
$ ssh -1 nomeusuario@nomemaquina.dominio.ext # Força SSH versão 1
$ ssh -1 -o RSAAuthentication=no -l nomeusuario foo.host
    # força senha em SSH1
$ ssh -o PreferredAuthentications=password -l nomeusuario foo.host
    # força senha em SSH2
```

Para o usuário, o `ssh` funciona como um `telnet` mais esperto e mais seguro (não vai explodir com ^]).

9.5.2 Redirecionamento de porta para tunelamento SMTP/POP3

Para estabelecer um pipe para conectar à porta 25 de um *servidor-remoto* a partir da porta 4025 do sistema local, e para a porta 110 do *servidor-remoto* a partir da porta 4110 do sistema local através do `ssh`, execute na máquina local:

```
# ssh -q -L 4025:servidor-remoto:25 4110:servidor-remoto:110 \
         nomeusuario@servidor-remoto
```

Essa é uma forma segura de fazer conexões com servidores SMTP/POP3 através da Internet. Ajuste a entrada `AllowTcpForwarding` para `yes` no arquivo `/etc/ssh/sshd_config` da máquina remota.

9.5.3 Conectando com menos senhas – RSA

Pode-se evitar de ter que lembrar de uma senha para cada sistema remoto usando RSAAuthentication (protocolo SSH1) ou PubkeyAuthentication (protocolo SSH2).

No sistema remoto, defina as respectivas entradas, "RSAAuthentication yes" ou "PubkeyAuthentication yes", em `/etc/ssh/sshd_config`.

Então gere chaves de autenticação localmente e instale a chave pública no sistema remoto:

```
$ ssh-keygen          # RSAAuthentication: chave RSA1 para SSH1
$ cat .ssh/identity.pub | ssh usuario1@remoto \
        "cat - >>.ssh/authorized_keys"
...
$ ssh-keygen -t rsa   # PubkeyAuthentication: chave RSA para SSH2
$ cat .ssh/id_rsa.pub | ssh usuario1@remoto \
        "cat - >>.ssh/authorized_keys"
...
$ ssh-keygen -t dsa   # PubkeyAuthentication: chave DSA para SSH2
$ cat .ssh/id_dsa.pub | ssh usuario1@remoto \
        "cat - >>.ssh/authorized_keys"
```

Pode-se mudar a frase-senha mais tarde com "`ssh-keygen -p`". Não esqueça de verificar as configurações testando a conexão. Em caso de qualquer problema, use "`ssh -v`".

Você pode adicionar opções para as entradas em `authorized_keys` para limitar as máquinas e executar comandos específicos. Veja `sshd(8)` para obter mais detalhes.

Note que SSH2 tem `HostbasedAuthentication`. Para isso funcionar, você precisa ajustar as configurações de `HostbasedAuthentication` para `yes` em ambos os `/etc/ssh/sshd_config` na máquina servidor e `/etc/ssh/ssh_config` ou `$HOME/.ssh/config` na máquina cliente.

9.5.4 Lidando com clientes SSH estranhos

Há alguns clientes SSH livres disponíveis para plataformas não semelhantes ao Unix.

Windows puTTY (`http://www.chiark.greenend.org.uk/~sgtatham/putty/`) (GPL)

Windows (cygwin) SSH em cygwin (`http://www.cygwin.com/`) (GPL)

Macintosh Classic macSSH (`http://www.macssh.com/`) (GPL) [Note que o Mac OS X inclui OpenSSH; use `ssh` na aplicação Terminal]

Veja também SourceForge.net, documentação do site (`http://www.sourceforge.net/docman/?group_id=1`), "6. CVS Instructions".

9.5.5 Configurando o `ssh-agent`

É mais seguro proteger sua chave de autenticação SSH com uma frase senha. Se isso não foi configurado, use `ssh-keygen -p` para configurar.

Coloque sua chave pública (por exemplo, `~/.ssh/id_rsa.pub`) no `~/.ssh/authorized_keys` em um host remoto usando conexão baseada em senha como descrito em 'Conectando com menos senhas – RSA' on the preceding page.

```
$ ssh-agent bash # ou ao invés, execute o programa zsh/tcsh/pdksh.
$ ssh-add ~/.ssh/id_rsa
Enter passphrase for /home/osamu/.ssh/id_rsa:
Identity added: /home/osamu/.ssh/id_rsa (/home/osamu/.ssh/id_rsa)
$ scp foo usuario@host.remoto:foo
 ... nenhuma frase senha necessária a partir daqui :-)
$^D
 ... terminando sessão ssh-agent
```

Para o servidor X, scripts de inicialização normais do Debian executam `ssh-agent` como processo pai. Então você só precisa executar `ssh-add` um vez.

Para mais detalhes, leia `ssh-agent(1)` e `ssh-add(1)`.

9.5.6 Resolução de problemas de SSH

Se você tiver problemas, verifique as permissões de arquivos de configuração e execute `ssh` com a opção ″`-v`“.

Use a opção ″`-P`“ se você for root e tiver problemas com um firewall; isso evita o uso das portas 1–1023 do servidor.

Se as conexões `ssh` para um sistema remoto pararem subitamente de funcionar, isso pode ser resultado de manutenção pelo administrador do sistema, normalmente uma mudança na `host_key` durante a manutenção do sistema. Depois de certificar-se de que é esse o caso e que ninguém está tentando simular o sistema remoto com algum hack esperto, pode-se conseguir uma conexão novamente removendo a entrada `host_key` de `$HOME/.ssh/known_hosts` na máquina local.

9.6 Correio

A configuração de correio se divide em três categorias:

- mail transfer agent (MTA) (agente de transferência de mensagens): `exim`, `postfix`, `sendmail`, `qmail`, `ssmtp`, `nullmailer`, ...
- utilitários de correio: `procmail`, `fetchmail`, `mailx`, ...
- mail user agent (MUA) (agente de correio de usuário): `mutt`, `emacs+gnus`, ...

9.6.1 Agentes de transporte de correio (MTAs)

Se quiser um MTA completo, use o `exim`. Referências:

- pacotes `exim-doc` e `exim-doc-html`
- `http://www.exim.org/`

O único MTA alternativo razoável é o `postfix`, se você se preocupa com a segurança. O `sendmail` e o `qmail` estão disponíveis como pacotes Debian mas não são recomendados.

Se você não precisar da capacidade de relay de um MTA como no caso de um sistema satélite como um PC laptop, você pode considerar usar um desses pacotes leves:

- `ssmtp`: precisa de conexão SMTP e tem capacidade de alias, ou
- `nullmailer`: pode enviar mas não tem capacidade de alias

Nesse momento, acho que o `exim` é o mais adequado mesmo para minha estação de trabalho pessoal, que é um PC laptop.

Você pode precisar remover o `exim` para a instalação de um desses pacotes conflitantes:

```
# dpkg -P --force-depends exim
# apt-get install nullmailer         # ou ssmtp
```

Smarthost

Se você está executando o `exim` em um sistema que está conectado através dos serviços voltados ao consumidor, por favor, certifique-se de enviar suas mensagens de saída através de um smarthost oferecido por seu ISP ou alguns outros. [6] Há algumas boas razões:

- para garantir novas tentativas do SMTP já que o smarthost de seu provedor geralmente tem uma conexão mais confiável.
- para evitar envio de mensagens diretamente a partir de um **endereço IP dinâmico** que serão muitas vezes bloqueados por listas de spam dial-up.
- para salvar sua largura de banda local para enviar mensagens com múltiplos destinatários.

As únicas exceções concebíveis são:

- a solução de emergência para um problema no serviço de SMTP do seu provedor.
- um experimento para propósitos educacionais.
- seu sistema se tornando um servidor profissional.

Configuração básica do Exim

Para usar o `exim` como seu MTA, configure o seguinte:

```
/etc/exim/exim.conf     "eximconfig" para criar e editar
/etc/inetd.conf         comente smtp para executar o exim como daemon
/etc/email-addresses    Adicione listas de endereços fontes falsos
verifique filtros usando exim -brw, -bf, -bF, -bV, ... etc.
```

Definindo um pega-tudo (catchall) para endereços de email inexistentes sob o Exim

No `/etc/exim/exim.conf` (Woody ou posterior), na parte DIRECTORS, adicione no fim (depois do direcionador localuser:) um direcionador pega-tudo que seleciona todos os endereços que o direcionador anterior não puder resolver (por Miquel van Smoorenburg):

```
catchall:
  driver = smartuser
    new_address = webmaster@meudominio.com
```

Se desejar ter uma receita mais detalhada para cada domínio virtual, etc, adicione o seguinte no fim de `/etc/exim/exim.conf` (por mim, não foi bem testado):

```
*@seudominio.com ${lookup{$1}lsearch*{/etc/email-addresses} \
        {$value}fail} T
```

---

[6] Você precisa seguir essa regra para quaisquer hosts em serviços de conexão discada, DSL, cabo, ou LAN através de algum roteador de banda larga. Mesmo se seu sistema doméstico tiver um IP fixo a partir de seu provedor (ISP), ainda é uma boa idéia seguir essa regra. Muitas estações de trabalho e servidores domésticos caem nessa categoria.

Então tenha uma entrada "*" em `/etc/email-addresses`.

Configurando reescrita seletiva de endereço para mensagens de saída sob o Exim

Pode-se fazer a reescrita seletiva de endereços para mensagens de saída para produzir cabeçalhos "From:" apropriados para o `exim` configurando próximo do fim de `/etc/exim/exim.conf`:

```
*@maquina1.algumacoisa.dnsdinam.org \
  "${if eq {${lookup{$1}lsearch{/etc/passwd}{1}{0}}} {1}  \
   {$0}{$1@algumacoisa.dnsdinam.org}}"  frFs
```

Isso reescreve todos os endereços combinando com `*@maquina1.algumacoisa.dnsdinam.org`.

1 Procura-se através de `/etc/password` para ver se a parte local ($1) é um usuário local ou não.

2 Se for um usuário local, o endereço é reescrito como a mesma coisa que era na primeira posição ($0).

3 Se não for um usuário local, é reescrito a parte do domínio.

Configurando autenticação SMTP sob o Exim

Alguns serviços SMTP como yahoo.com requerem autenticação SMTP. Configure o arquivo `/etc/exim/exim.conf` como a seguir:

```
remote_smtp:
  driver = smtp
  authenticate_hosts = smtp.mail.yahoo.com
...

smarthost:
  driver = domainlist
  transport = remote_smtp
  route_list = "* smtp.mail.yahoo.com bydns_a"
...

plain:
  driver = plaintext
  public_name = PLAIN
  client_send = "^cmatheson3^this_is_my_password"
```

Não esqueça das aspas na última linha.

9.6.2 Baixando mensagens – Fetchmail

O `fetchmail` é executado em modo daemon para baixar as mensagens de uma conta POP3 com um ISP para o sistema de correio local. Configure:

```
/etc/init.d/fetchmail
/etc/rc?.d/???fetchmail executa update-rc.d fetchmail com prioridade padrão 30
/etc/fetchmailrc        arquivo de configuração (chown 600, de propriedade de fetc
```

As informações sobre como iniciar o `fetchmail` como um daemon a partir do script `init.d` para o Potato são confusas (no Woody isso foi resolvido). Veja os arquivos de exemplo `/etc/init.d/fetchmail` e `/etc/fetchmailrc` nos scripts de exemplo (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

Se seus cabeçalhos das mensagens estiverem contaminados com ^M devido ao servidor de correio de seu ISP, adicione ″stripcr″ às suas opções em `$HOME/.fetchmailrc`:

```
options fetchall no keep stripcr
```

9.6.3 Processando mensagens – Procmail

O `procmail` é um programa de entrega de correio local e de filtro. É necessário criar `$HOME/.procmailrc` para cada conta que usá-lo. Exemplo: _procmailrc (`http://www.debian.org/doc/manuals/debian-reference/examples/`)

9.6.4 Lendo mensagens – Mutt

Use o `mutt` como o agente de correio do usuário (MUA) combinado com o `vim`. Personalize com `~/.muttrc`; por exemplo:

```
# usar modo visual e "gq" para reformatar citações
set editor="vim -c 'set tw=72 et ft=mail'"
#
# supressão de cabeçalho tomada do manual (Sven's Draconian header weeding)
#
ignore *
unignore from: date subject to cc
unignore user-agent x-mailer
hdr_order from subject to cc date user-agent x-mailer
auto_view application/msword
....
```

Adicione o seguinte ao `/etc/mailcap` ou `$HOME/.mailcap` para exibir mensagens em HTML e anexos MS Word em seguida:

```
text/html; lynx -force_html %s; needsterminal;
application/msword; /usr/bin/antiword '%s'; copiousoutput;
description="Microsoft Word Text"; nametemplate=%s.doc
```

9.7 Localização

O Debian é internationalizado, oferecendo suporte a um número crescente de idiomas e convenções de uso locais. A próxima subseção lista algumas das formas de diversidade que o Debian suporta atualmente, e as subseções seguintes discutem a **localização**, o processo de personalização de seu ambiente de trabalho para permitir entrada e saída atuais de seu(s) idioma(s) escolhido(s) e convenções para datas, formatos numéricos e monetários, e outros aspectos de um sistema que variam de acordo com sua região.

9.7.1 Básico da localização

Há vários detalhes para a personalização da localização e suporte a idioma nativo.

Localizando o teclado

O Debian é distribuído com mapas de teclado para aproximadamente duas dúzias de teclados. No Woody, reconfigure o teclado com:

- `dpkg-reconfigure --priority=low console-data # console`
- `dpkg-reconfigure --priority=low xserver-xfree86 # XF4`
- `dpkg-reconfigure --priority=low xserver-common-v3 # XF3`

Localizando arquivos de dados

A grande maioria dos pacotes de software do Debian suporta manipulação de caracteres não-US-ASCII através da variável de ambiente LC_CTYPE oferecida pela tecnologia **locale** na glibc.

- 8-bit limpo: praticamente todos os programas
- outros conjuntos de caracteres Latinos (por exemplo ISO-8859-1 ou ISO-8859-2): a maioria dos programas
- idiomas multi-byte como Chinês, Japonês, ou Coreano: muitas aplicações novas

Localizando a exibição

O X pode exibir qualquer codificação, incluindo UTF-8, e suporta todas as fontes. A lista inclui não apenas todas as fontes de 8 bits mas também fontes de 16 bits tais como Chinês, Japonês, ou Coreano. O método de entrada de caracter multi-byte é suportado pelo mecanismo 'Métodos de entrada X alternativos' on page 179. Veja 'Exemplo para um sistema X window multi-idiomas' on page 176 e 'Exemplo para UTF-8 no X' on page 180.

A exibição código Japonês EUC também é disponível em console gráfico (S)VGA através do pacote `kon2`. Há um novo display Japonês alternativo, `jfbterm` que usa um console framebuffer, também. Nesses ambientes de console, o método de entrada Japonês precisa ser fornecido pela aplicação. Use o pacote `egg` para o Emacs e use o pacote `jvim` ″japonesado″ para o ambiente Vim.

A instalação de fontes não Unicode no X ajudará na exibição de documentos com qualquer codificação no X. Então não se preocupe muito com a codificação das fontes.

Localizando mensagens e documentação

Existem traduções para muitas mensagens e documentos que são exibidos no sistema Debian, tais como mensagens de erro, saída de programa padrão, menus, e páginas de manual. Atualmente, existe suporte para páginas de manual em Alemão, Espanhol, Finlandês, Francês, Húngaro, Italiano, Japonês, Coreano, Polonês, Português, Chinês e Russo através dos pacotes `manpages-`*`LANG`* (onde *LANG* é uma lista separada por vírgulas de códigos de país ISO de duas letras. Use `apt-cache search manpages- |less` para obter uma lista das páginas de manual Unix disponíveis.)

Para acessar uma página de manual NLS, o usuário precisa definir a variável de ambiente LC_MESSAGES para o valor apropriado. Por exemplo, no caso das páginas de manual da língua italiana, LC_MESSAGES precisa ser configurada para `it`. O programa `man` então procurará por páginas de manual italianas sob `/usr/share/man/it/`.

9.7.2 Locales

O Debian suporta a tecnologia **locale**. Locale é um mecanismo que permite que um programa forneça saída apropriada e funcionalidade de acordo com as convenções locais como conjunto de caracteres, formato de data e hora, símbolo da moeda, e outros. Ele usa variáveis de ambiente para determinar o comportamento apropriado. Por exemplo, considerando que você tenha ambos os locales Inglês Americano e Alemão instalados em seu sistema, as mensagens de erro de muitos programas podem ser multi-idiomas:

```
$ LANG="en_US" cat foo
cat: foo: No such file or directory
$ LANG="de_DE" cat foo
cat: foo: Datei oder Verzeichnis nicht gefunden
```

A Glibc oferece suporte a essa funcionalidade como uma biblioteca. Veja `locale(7)`.

9.7.3 Introdução a locales

A descrição completa de locale consiste de 3 partes: `xx_YY.ZZZZ`.

- **`xx`**: códigos de idioma ISO 639 (minúsculas)

- `YY`: códigos de país ISO 3166 (maiúsculas)
- `ZZZZ`: codeset, i.e., conjunto de caracteres ou identificador de codificação.

Para códigos de idiomas e códigos de países, veja a descrição pertinente no `info gettext`.

Por favor, note que essa parte de codeset pode ser normalizada internamente para obter compatibilidade sobre plataformas removendo todos os − e convertendo todos os caracteres para minúsculas. Codesets típicos são:

- **UTF-8**: Unicode para todas as regiões, normalmente em 1-3 Octetos (novo padrão de facto)
- **ISO-8859-1**: Europa ocidental (padrão de facto antigo)
- **ISO-8859-2**: Europe oriental (Bósnio, Croata, Tcheco, Húngaro, Polonês, Romeno, Sérvio, Eslovaco, Eslovênio)
- **ISO-8859-3**: Maltês
- **ISO-8859-5**: Macedônio, Sérvio
- **ISO-8859-6**: Árabe
- **ISO-8859-7**: Grego
- **ISO-8859-8**: Hebreu
- **ISO-8859-9**: Turco
- **ISO-8859-11**: Thai (=TIS-620)
- **ISO-8859-13**: Letão, Lituano, Maori
- **ISO-8859-14**: Galês (Welsh)
- **ISO-8859-15**: Europe ocidental com euro
- **KOI8-R**: Russo
- **KOI8-U**: Ucraniano
- **CP1250**: Tcheco, Húngaro, Polonês (original do MS Windows)
- **CP1251**: Búlgaro, Bielorrusso (original do MS Windows)
- **eucJP**: Japonês estilo Unix (=ujis)
- **eucKR**: Coreano estilo Unix
- **GB2312**: Chinês Simplificado estilo Unix (=GB, =eucCN) para zh_CN
- **Big5**: Chinês Tradicional para zh_TW
- **sjis**: Japonês estilo Microsoft (Shift-JIS)

Sobre o significado de jargões básicos de sistema de codificação:

- **ASCII**: 7 bits (0-0x7f)
- **ISO-8859-?**: 8 bits (0-0xff)
- **ISO-10646-1**: Conjunto de Caracteres Universal (UCS) (31 bits, 0-0x7fffffff)
- **UCS-2**: Primeiros 16 bit do UCS como 2 Octetos seguidos (Unicode: 0-0xffff)
- **UCS-4**: UCS como 4 Octetos seguidos (UCS: 0-0x7fffffff)
- **UTF-8**: UCS codificado em 1-6 Octetos (normalmente em 3 Octetos)
- **ISO-2022**: 7 bits (0-0xff) com a seqüência de escape. ISO-2022-JP é a codificação mais popular para o e-mail japonês.
- **EUC**: combinação 8 bits + 16 bits (0-0xff), estilo Unix
- **Shift-JIS**: combinação 8 bits + 16 bits (0-0xff), estilo Microsoft.

ISO-8859-?, EUC, ISO-10646-1, UCS-2, UCS-4, e UTF-8 compartilham o mesmo código com ASCII para os caracteres de 7 bit. EUC ou Shift-JIS usam caracteres de bit alto (0x80-0xff) para indicar que parte da codificação é 16 bit. UTF-8 também usa caracteres de bit alto (0x80-0xff) para indicar bytes de seqüência de caracter de não 7 bit e essa é a forma mais inteligente de um sistema de codificação lidar com caracteres não-ASCII.

Por favor, note a diferença de ordem de byte da implementação Unicode:

- **UCS-2, UCS-4 padrões**: big endian
- **UCS-2, UCS-4 Microsoft**: little endian para ix86 (dependente de máquina)

Para mais detalhes veja Introdução a i18n (`http://www.debian.org/doc/manuals/intro-i18n/`).

9.7.4 Ativando suporte a locale

O Debian **não** vem com todos os locales disponíveis pré-compilados. Verifique `/usr/lib/locale` para ver quais locales (além do padrão "C") estão compilados para seu sistema. Se o que você precisar não estiver presente, você tem duas opções:

- Editar o `/etc/locale.gen` para adicionar o locale desejado, e então executar `locale-gen` como root para compilá-lo. Veja `locale-gen(8)` e páginas de manual listadas em sua seção "SEE ALSO".
- Executar `dpkg-reconfigure locales` para reconfigurar o pacote `locales` . Ou se ele ainda não estiver instalado, instalar `locales` chamará a interface debconf para permitir-lhe a escolha dos locales necessários e compilar a base de dados.

9.7.5 Ativando um locale específico

As seguintes variáveis de ambiente são avaliadas nessa ordem para fornecer valores particulares de locale para os programas:

1. LANGUAGE: Essa variável de ambiente consiste de uma lista de nomes de locale em ordem de prioridade. Usada somente se o locale POSIX estiver definido para um valor diferente de "C" [no Woody; na versão Potato sempre tinha prioridade sobre o locale POSIX]. (extensão GNU)

2. LC_ALL: Se for não nula, o valor é usado para todas as categorias de locale. (POSIX.1) Geralmente "" (null).

3. LC_*: Se for não nula, seu valor é usado para a categoria correspondente (POSIX.1). Geralmente "C".

   As variáveis LC_* são:

   - LC_CTYPE: Classificação de caracteres e conversão maiúsculas/minúsculas.
   - LC_COLLATE: Ordem de comparação.
   - LC_TIME: Formatos de data e hora.
   - LC_NUMERIC: Formatos numéricos não monetários.
   - LC_MONETARY: Formatos monetários.
   - LC_MESSAGES: Formatos de mensagens informativas e de diagnóstico e respostas interativas.
   - LC_PAPER: Tamanho de papel.
   - LC_NAME: Formatos de nome.

- LC_ADDRESS: Formatos de endereço e informação de localização.
- LC_TELEPHONE: Formatos de número de telefone.
- LC_MEASUREMENT: Unidades de medidas (Métricas ou Outras).
- LC_IDENTIFICATION: Metadado sobre informação de locale.

4 LANG: Se for não nula e LC_ALL for indefinida, o valor é usado para todas as categorias LC_* de locale com valores indefinidos. (POSIX.1) Geralmente é "C".

Note que algumas aplicações (por exemplo, Netscape 4) ignoram configurações LC_*.

O programa `locale` pode exibir as configurações ativas de locale e os locales disponíveis; veja `locale(1)`. (NOTA: `locale -a` lista todos os locales que seu sistema conhece; isso *não* significa que todos eles estão compilados! Veja 'Ativando suporte a locale' on the facing page.)

9.7.6 Locale de formato de data ISO 8601

O suporte de locale para o padrão internacional de data de `yyyy-mm-dd` (formato de data ISO 8601) é fornecido pelo locale chamado `en_DK`, "Inglês na Dinamarca", o que é um pouco de brincadeira :-) Isso parece funcionar apenas em tela de console para o `ls`.

9.7.7 Exemplo para os Estados Unidos (ISO-8859-1)

Adicione as seguintes linhas ao `~/.bash_profile`:

```
LC_CTYPE=en_US.ISO-8859-1
export LC_CTYPE
```

9.7.8 Exemplo para a França com o sinal de Euro (ISO-8859-15)

Adicione as seguintes linhas ao `~/.bash_profile`:

```
LANG=fr_FR@euro
export LANG
LC_CTYPE=fr_FR@euro
export LC_CTYPE
```

Configure o teclado para o "AZERTY" Francês como descrito em 'Localizando o teclado' on page 171; adicione as páginas de manual em francês instalando `manpages-fr`. A tecla Alt-direito nos Estados Unidos é chamada Alt-Gr na Europa. Pressioná-lo junto com algumas teclas cria vários caracteres acentuados e caracteres especiais. Por exemplo, Alt-Gr+E cria um sinal de Euro.

Muitas línguas da Europa ocidental podem ser configuradas similarmente.

Veja o Debian Euro HOWTO (`http://www.debian.org/doc/manuals/debian-euro-support/`) para adicionar suporte para a nova moeda Euro e Utiliser et configurer Debian pour le français (`http://www.debian.org/doc/manuals/fr/debian-fr-howto/`) para mais detalhes em francês.

9.7.9 Exemplo para um sistema X window multi-idiomas

Vamos configurar um sistema X window multi-idioma que suporta simultaneamente japonês, inglês, alemão e francês com as codificações EUC, UTF-8 e ISO-8859-1 em diferentes consoles.

Eu mostrarei a você uma personalização usando o sistema de menus do Debian. Veja os detalhes do sistema de menus do Debian em `/usr/share/doc/menu/html/index.html`. Também criarei um atalho para o navegador web `mozilla` nesse exemplo. [7]

- adicione um suporte para o locale ja_JP.eucJP Japonês e os outros locales requeridos usando o método descrito em 'Localização' on page 171. (para todos)
- instale o sistema de conversão Kana-para-Kanji e dicionário (para japonês):
  - `canna` – Servidor local (licença "cerveja grátis" (free-beer)), ou
  - `freewnn-jserver` – Servidor extensível por rede (Domínio Público)
- instale o sistema de método de entrada Japonês (para japonês):
  - `kinput2-canna` – para o X, ou
  - `kinput2-canna-wnn` – para o X, e
  - `egg` – funciona diretamente com Emacsen mesmo em console (opcional)
- Instale terminais compatíveis (para todos):
  - `xterm` – X (para ISO-8859-1 and UTF-8),
  - `kterm` – X (para Japonês EUC), e
  - `mlterm` – X (multi-idioma).
- adicione todos os pacotes de fontes requeridos. (para todos)
- crie um `~/.xsession` que defina um ambiente X específico do usuário como descrito em 'Sessões X personalizadas' on page 153 (para todos):

```
#!/bin/sh
# Isso faz o X funcionar quando uso su para root.
if [ -z "$XAUTHORITY" ]; then
        XAUTHORITY=$HOME/.Xauthority
        export XAUTHORITY
fi
```

---

[7] Nesse exemplo, são desenvolvidos 2 desvios de bugs para a versão do `blackbox` em 2003. Eu uso `sh -c` em comando. Além disso a entrada `~/.menu/*` não é usada mas ao invés disso foi usada `/etc/menu/*` que requer root.

```
# Definir ambiente específico através do sistema de menus debian.
# Redefinir locale
unset LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_MESSAGES
unset LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
unset LC_IDENTIFICATION LC_ALL LANG LANGUAGE PAGER
# ajustar padrão de locale no X
LANG=C
# exportar locale
export LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_MESSAGES
export LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
export LC_IDENTIFICATION LC_ALL LANG LANGUAGE PAGER
###
# activar método de entrada para japonês com kinput2
kinput2 &
XMODIFIERS="@im=kinput2"
export XMODIFIERS
# Sobre o gerenciador de janelas blackbox (leve)
exec blackbox
#exec xfwm
#exec wmaker
```

- ajuste o locale em `~/.bash_profile` para consoles Linux (para todos).
- remova todos os ajustes de locale em `~/.bashrc`, se existiram (para todos).
- crie alguns arquivos em `/etc/menu/` (para todos).
  - `/etc/menu/xterm-local`: (adicione novas entradas ao menu) [8]

    ```
    ?package(xterm):\
     needs=x11\
     section=XShells\
     longtitle="XTerm: emulador de terminal (en_US.ISO-8859-1)"\
     title="XTerm (en_US.ISO-8859-1)"\
     command="sh -c ’LC_ALL=en_US.ISO-8859-1 xterm’"
    ?package(xterm):\
     needs=x11\
     section=XShells\
     longtitle="XTerm: emulador de terminal (de_DE.ISO-8859-1)"\
     title="XTerm (de_DE.ISO-8859-1)"\
     command="sh -c ’LC_ALL=de_DE.ISO-8859-1 xterm -T xterm-de’"
    ?package(xterm):\
     needs=x11\
     section=XShells\
     longtitle="XTerm: emulador de terminal para o X com suporte Unicode (J
     title="UXTerm (ja_JP.UTF-8)"\
     command="sh -c ’LC_ALL=ja_JP.UTF-8 uxterm’"
    ```

---

[8] Use um nome de arquivo que não conflite com quaisquer nomes de pacotes.

  - `/etc/menu/kterm`: (sobrepõe o padrão do sistema) [9]

    ```
    ?package(kterm):\
     needs="x11"\
     section="XShells"\
     command="sh -c 'LC_ALL=ja_JP.eucJP PAGER=w3m /usr/X11R6/bin/kterm -xi
     title="Kanji Terminal"
    ?package(kterm):\
     needs="x11"\
     section="XShells"\
     command="sh -c 'LANG=ja_JP.eucJP \
              LC_MESSAGES=en_US.ISO-8859-1 PAGER=w3m /usr/X11R6/bin/kterm -
     title="Terminal Kanji (bilíngüe)"
    ```

  - `/etc/menu/mozilla-local`: (adicionar um novo atalho) [10]

    ```
    ?package(mozilla-browser):needs="x11" section="/" \
            title=" Navegador Mozilla" command="mozilla-1.5" hints="Navega
            icon=/usr/share/pixmaps/mozilla.xpm
    ```

  - execute `update-menus` a partir da conta root.

- adicione as seguintes linhas ao arquivo `~/.muttrc` (para japonês):

  ```
  # Suporte a UTF-8 não é popular no ambiente EMACS japonês
  # codificação de 7-bit iso-2022-jp é mais fácil para todos.
  # ordem de codificação padrão = us-ascii --> iso-8859-1 --> iso-2022-jp
  set send_charset="us-ascii:iso-8859-1:iso-2022-jp"
  set allow_8bit=no
  ```

- ative XIM `kinput2` para aplicações X (para japonês)
  - adicione `*inputMethod: kinput2` e `KTerm*VT100*OpenIm: true` em seu arquivo de recursos X, `~/.Xresources` (parece que o Debian cuida disso automaticamente de alguma forma).
  - Algumas aplicações (como `mlterm`) também permitem que você configure `*inputMethod:` e outras informações dinamicamente durante a execução (pressione **Ctrl-BotãoMouse-3** no `mlterm`).

- inicie o X digitando `startx` ou a partir de um dos gerenciadores de display (`xdm`, `gdm`, `kdm`, `wdm`, . . .) (para todos).

- inicie uma aplicação compatível com japonês como Vim 6, (x)emacs21, mc-4.5, mutt-1.4, . . . no `kterm` (para japonês). (Emacs parece ser a plataforma mais popular, apesar de eu não usá-lo.)

- pressione **Shift+Espaço** para alternar entre modo de entrada de caracter japonês ligado e desligado (para japonês).

---

[9] Use um nome de arquivo que sobrepõe (overwrap) o nome do pacote.

[10] A barra em `section=”/‟` habilita entrada no menu inicial, e o espaço no começo em `title=” Navegador Mozilla‟` habilita entrada no topo da lista.

- leia a página de manual localizada iniciando o comando em um console localizado (para todos).

Para outros suportes a idioma CJK, veja as seções seguintes e as páginas da SuSE para CJK (`http://www.suse.de/~mfabian/suse-cjk/suse-cjk.html`).

9.7.10 Métodos de entrada X alternativos

Há muitos pacotes de suporte de métodos de entrada do X alternativos disponíveis:

Idioma	LC_CTYPE	serv. XIM	XMODIFIERS	Tecla início
Japonês	ja_JP*	kinput2	"@im=kinput2"	Shift-Espaço
Coreano	ko_KR*	ami	"@im=Ami"	Shift-Espaço
Chinês(T)	zh_TW.Big5	xcin	"@im=xcin-zh_TW.big5"	Ctrl-Espaço
Chinês(S)	zh_CN.GB2312	xcin	"@im=xcin-zh_CN.GB2312"	Ctrl-Espaço

O método de entrada japonês `kinput2` é oferecido pelos pacotes como `kinput2-canna-wnn`, `kinput2-canna`, e `kinput2-wnn`. O japonês precisa de um servidor de dicionário tal como `canna` e `freewn-jserver` para ser prático.

9.7.11 Emuladores de terminal no X

Há muitos consoles que suportam codificações de 8 bits simples quando os pacotes de fonte pertinentes são instalados:

- `xterm` – O emulador de terminal X
- `gnome-terminal` – `xterm` para o Gnome
- `konsole` – `xterm` para o KDE
- `rxvt` – terminal VT102 (mais leve)
- `aterm` – terminal VT102 para o WM Afterstep
- `eterm` – terminal VT102 para o WM Enlightenment
- `wterm` – terminal VT102 para o WM WindowMaker

Os suportes de codificação multi-byte dos consoles X são fornecidos pelo `xterm` através da codificação UTF-8 ('Exemplo para UTF-8 no X' on the following page). Outros suportes de codificação tradicionais estão em progresso (em 2003). Os seguintes pacotes oferecem suportes a codificações tradicionais:

- `aterm-ml` – Multi-idioma
- `kterm` – Multi-idioma (Japonês, . . .)
- `rxvt-ml` – Multi-idioma
- `wterm-ml` – Multi-idioma
- `cxterm-big5` – Chinês (Trad., Big5)
- `cxterm-gb` – Chinês (Simp., GB)
- `cxterm-ks` – Chinês (KS)
- `cxterm-jis` – Japonês

- `hanterm-classic` – Coreano (Hangul)
- `hanterm-xf` – Coreano (Hangul)
- `hztty` – Chinês (GB, Big5, zW/HZ)

Para o `kterm` (e possivelmente outros), você pode desejar ativar XIM através do menu depois da ação do mouse Ctrl-click-botão_do_meio.

9.7.12 Exemplo para UTF-8 no X

O suporte a UTF-8 para o emulador de terminal X é fornecido pelo programa `uxterm` no pacote `xterm` para o XFree86 4.x. Ele habilita suporte para todos os idiomas. Ele é um wrapper sobre o programa `xterm(1)` que invoca esse último com a classe de recursos X "UXterm" definida.

Por exemplo, para habilitar uma bela exibição grande de caracteres de inglês, russo, japonês, chinês e coreano, adicione o seguinte ao seu arquivo `~/.Xresources` depois de instalar todas as fontes pertinentes:

```
! definindo fonte grande
UXTerm*font: -misc-fixed-medium-r-normal-*-18-120-100-100-c-90-iso10646-1
! Usar XIM para Japonês
*inputMethod: kinput2
```

Então execute `xrdb -merge ~/.Xresources` para atualizar os recursos X como descrito em 'Recursos do X' on page 157.

Entretanto, muitos dos pacotes de programas de console populares como o `vim`, `mutt`, e `emacs` tornaram-se compatíveis com UTF-8 recentemente (Woody-Sarge). Programa como o `mc` ainda não é compatível com UTF-8 mas é simplesmente 8 bits limpos. Se você está editando com a parte de 7 bit ASCII de um arquivo de codificação desconhecida ou mista, é mais seguro usar o editor de 8 bits limpos sem locale.

Veja The Unicode HOWTO (`http://www.tldp.org/HOWTO/Unicode-HOWTO.html`).

9.7.13 Exemplo para UTF-8 em um console framebuffer

O suporte UTF-8 em um console FB é fornecido pelo `bterm` e é usado no `debian-installer`.

9.7.14 Além dos locales

Quando você estiver configurando o sistema para um **ambiente de idioma nativo** pela primeira vez, por favor, pense em usar o `tasksel` ou o `aptitude` para determinar quais pacotes são selecionados ao escolher a tarefa de ambiente de idioma correspondente. É útil fazer escolha de pacote mesmo para a configuração multi-idioma. Se você encontrar algum conflito de dependência de pacote durante a instalação de seu sistema configurado cuidadosamente, evite instalar qualquer software que conflite com o sistema existente. Você pode ter que usar o `update-alternative` para obter o estado original novamente para alguns comandos, já que um recém instalado pode ter uma prioridade mais alta que os já existentes.

Os maiores e mais novos programas estão usando a glibc 2.2 e muitas vezes são internacionalizados. Então uma versão especialmente localizada como o `jvim` para o Vim não deve ser necessária já que sua funcionalidade é oferecida pela versão 6.0 do `vim` no X. Na verdade, ainda está um pouco grosseiro. Já que o `jvim` tem uma versão compilada com suporte de método de entrada japonês direto (`canna`) mesmo no console e resolve muitos outros problemas específicos do Japonês de forma madura, você pode ainda querê-lo :-)

Os programas podem precisar ter configuração além do `locale` para se ter um ambiente de trabalho confortável. O pacote `language-env` e seu comando `set-language-env` facilita enormemente esse processo.

Veja também o documento de internacionalização, Introdução a i18n (`http://www.debian.org/doc/manuals/intro-i18n/`). Ele é dedicado aos desenvolvedores mas também é útil para os administradores de sistema.

Capítulo 10

Configuração de rede

Este capítulo está focado em administração de rede no Debian. Para uma introdução geral a redes no GNU/Linux, leia o Net-HOWTO (`http://www.tldp.org/HOWTO/Net-HOWTO/index.html`).

Para um sistema Debian ser capaz de acessar a Internet, suas interfaces de rede precisam ser suportadas pelo kernel e estar configuradas apropriadamente.

O primeiro requisito é o suporte de kernel para os dispositivos de interface de rede como placas Ethernet, placas Wi-Fi e modems. Para obter esse suporte você pode precisar recompilar o kernel ou adicionar módulos a ele como descrito em 'O kernel Linux no Debian' on page 101.

A configuração de dispositivos de rede é explicada abaixo. A informação nesse capítulo foi atualizada para o Sarge. Muito dela não se aplica para versões anteriores.

10.1 Básico de rede IP

Um sistema Debian pode ter várias interfaces, cada uma com um endereço de Protocolo de Internet (IP) diferente. As interfaces podem ser de diferentes tipos, incluindo:

- Loopback: `lo`
- Ethernet: `eth0,eth1,...`
- Wi-Fi: `wlan0,wlan1,...` [1]
- Token Ring: `tr0,tr1,...`
- PPP: `ppp0,ppp1,...`

Há uma grande faixa de outros dispositivos de rede disponíveis, incluindo SLIP, PLIP (IP em linha serial e paralela), dispositivos "shaper" para controlar o tráfego em certas interfaces, frame relay, AX.25, X.25, ARCnet, e LocalTalk.

---

[1] Note que uma interface Wi-Fi é na verdade um apelido (alias) para uma interface Ethernet que dá acesso aos parâmetros de configuração peculiares ao Wi-Fi. Esses parâmetros são controlados usando o programa `iwconfig`.

Toda interface de rede conectada diretamente à Internet (ou a qualquer rede baseada em IP) é identificada por um endereço IP de 32 bit único. [2] O endereço IP pode ser dividido na parte que endereça a rede e na parte que endereça o sistema (host). Se você tiver um endereço IP, defina para 1 os bits que são parte do endereço de rede e defina para 0 os bits que são parte do endereço do sistema e então você obtém a máscara da rede.

Tradicionalmente, redes IP foram agrupadas em classes cujas partes do endereço de rede foram de comprimento 8, 16 ou 24 bits. [3]

	endereços IP	máscara rede	comprimento
Classe A	1.0.0.0 - 126.255.255.255	255.0.0.0	= /8
Classe B	128.0.0.0 - 191.255.255.255	255.255.0.0	= /16
Classe C	192.0.0.0 - 223.255.255.255	255.255.255.0	= /24

Endereços IP fora dessas faixas são usados para propósitos especiais.

Há faixas de endereços em cada classe reservados para uso em redes locais (LANs). Esses endereços têm a garantia de não conflitar com nenhum endereço no domínio da Internet. (Pelo mesmo motivo, se um desses endereços é designado para um sistema então esse sistema não pode acessar a Internet diretamente, precisa acessá-la através de um gateway que atua como um proxy para serviços individuais ou que faça Translação de Endereço de Rede (Network Address Translation).) Essas faixas de endereços são dadas na tabela a seguir junto com o número de faixas em cada classe.

	endereços de rede	compr.	quantidade
Classe A	10.x.x.x	/8	1
Classe B	172.16.x.x - 172.31.x.x	/16	16
Classe C	192.168.0.x - 192.168.255.x	/24	256

O primeiro endereço em uma rede IP é o endereço da própria rede. O último endereço é o endereço de broadcast da rede. [4] Todos os outros endereços podem ser alocados a sistemas na rede. Desses, o primeiro ou o último endereço normalmente é alocado para o gateway de Internet para a rede.

A tabela de roteamento contém a informação do kernel sobre como enviar pacotes IP aos seus destinos. Aqui está uma amostra de tabela de roteamento para um sistema Debian em uma rede local (LAN) com endereço IP 192.168.50.x/24. O sistema 192.168.50.1 (também na LAN) é um roteador para a rede corporativa 172.20.x.x/16 e o sistema 192.168.50.254 (também na LAN) é um roteador para a Internet para todos.

---

[2] Isso é verdadeiro se IP versão 4 estiver sendo usado. Em endereços IPv6 são 128 bits. Veja `http://www.ipv6.org/`.

[3] Esse sistema era inflexível e desperdiçava muitos endereços IP, então hoje as redes IPv4 são alocadas com partes de endereço de rede de comprimento variável.

[4] O endereço da rede pode ser obtido fazendo uma operação lógica E (AND) bit-a-bit entre o endereço na rede com a máscara de rede. O endereço de broadcast pode ser obtido fazendo uma operação lógica OU (OR) bit-a-bit entre o endereço da rede e o complemento de 1 da máscara de rede.

```
# route
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref Use Iface
127.0.0.0      *               255.0.0.0       U     0      0     2 lo
192.168.50.0   *               255.255.255.0 U     0      0   137 eth0
172.20.0.0     192.168.50.1    255.255.0.0     UG    1      0     7 eth0
default        192.168.50.254 0.0.0.0          UG    1      0    36 eth0
```

- A primeira linha depois do cabeçalho diz que o tráfego destinado à rede `127.x.x.x` será roteado para `lo` a interface de loopback.
- A segunda linha diz que o tráfego destinado aos sistemas na LAN serão roteados através de `eth0`.
- A terceira linha diz que o tráfego destinado à rede corporativa será roteado para o gateway `192.168.50.1` e também através de `eth0`.
- A quarta linha diz que o tráfego destinado à Internet será roteado para o gateway `192.168.50.254` e também através de `eth0`.

Os endereços IP na tabela também podem aparecer como nomes que são obtidos procurando endereços em `/etc/networks` ou usando o resolvedor da Biblioteca C.

Além de rotear, o kernel pode fazer translação de endereço de rede, shaping, e filtro.

Veja o Net-HOWTO (`http://www.tldp.org/HOWTO/Net-HOWTO/index.html`) e outros HOWTOs de rede (`http://www.tldp.org/HOWTO/Networking-Overview-HOWTO.html`) para informações mais aprofundadas.

10.2 Configuração de rede de baixo nível

As ferramentas de configuração de rede de baixo nível tradicionais nos sistemas GNU/Linux são os programas `ifconfig` e `route` que vêm no pacote `net-tools`. Essas ferramentas oficialmente foram superadas pelo `ip` que vem no pacote `iproute`. O programa `ip` funciona no Linux 2.2 e superiores e é mais capaz que as ferramentas antigas. Entretanto, as ferramentas antigas ainda funcionam e são mais familiares a muitos usuários.

10.2.1 Configuração de rede de baixo nível – `ifconfig` e `route`

Aqui está uma ilustração de como mudar o endereço IP da interface `eth0` de `192.168.0.3` para `192.168.0.111` e tornar a `eth0` como rota para a rede `10.0.0.0` via `192.168.0.1`. Nós começamos executando `ifconfig` e `route` sem argumentos de interface para exibir o estado atual de todas as interfaces de rede e roteamento.

```
# ifconfig
eth0 Link encap:Ethernet  HWaddr 08:00:46:7A:02:B0
     inet addr:192.168.0.3  Bcast:192.168.0.255  Mask:255.255.255.0
     UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
     RX packets:23363 errors:0 dropped:0 overruns:0 frame:0
     TX packets:21798 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:100
     RX bytes:13479541 (12.8 MiB)  TX bytes:20262643 (19.3 MiB)
     Interrupt:9

lo   Link encap:Local Loopback
     inet addr:127.0.0.1  Mask:255.0.0.0
     UP LOOPBACK RUNNING  MTU:16436  Metric:1
     RX packets:230172 errors:0 dropped:0 overruns:0 frame:0
     TX packets:230172 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:0
     RX bytes:22685256 (21.6 MiB)  TX bytes:22685256 (21.6 MiB)
# route
Kernel IP routing table
Destination  Gateway      Genmask          Flags Metric Ref Use Iface
192.168.0.0  *            255.255.0.0      U     0      0     0 eth0
default      192.168.0.1  255.255.255.255  UG    0      0     0 eth0
```

Primeiro nós desativamos a interface.

```
# ifconfig eth0 inet down
# ifconfig
lo   Link encap:Local Loopback
  ... (agora sem a entrada eth0)
# route
  ... (agora sem entradas de roteamento na tabela)
```

Então nós a ativamos com o novo endereço IP e novo roteamento.

```
# ifconfig eth0 inet up 192.168.0.111 \
         netmask 255.255.0.0 broadcast 192.168.255.255
# route add -net 10.0.0.0 netmask 255.0.0.0 gw 192.168.0.1 dev eth0
```

O resultado:

```
# ifconfig
eth0 Link encap:Ethernet  HWaddr 08:00:46:7A:02:B0
     inet addr:192.168.0.111  Bcast:192.168.255.255  Mask:255.255.0.0
     UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
     ...

lo   Link encap:Local Loopback
     inet addr:127.0.0.1  Mask:255.0.0.0
```

```
    ...
# route
Kernel IP routing table
Destination  Gateway      Genmask      Flags Metric Ref Use Iface
192.168.0.0  *            255.255.0.0  U     0      0     0 eth0
10.0.0.0     192.168.0.1  255.0.0.0    UG    0      0     0 eth0
```

Para mais informações veja `ifconfig(8)` e `route(8)`.

10.2.2 Configuração de rede de baixo nível – `ip`

Os equivalentes para o `ip` dos comandos `ifconfig` e `route` anteriores são:

- `ip link show`
- `ip route list`
- `ip link set eth0 down`
- `ip addr del dev eth0 local 192.168.0.3`
- `ip addr add dev eth0 local 192.168.0.111/16 broadcast 192.168.255.255`
- `ip link set eth0 up`
- `ip route add dev eth0 to 10.0.0.0/8 src 192.168.0.111 via 192.168.0.1`

O programa `ip` mostra sua sintaxe de comando quando executado com o argumento `help`. Por exemplo, `ip link help` mostra:

```
Usage: ip link set DEVICE { up | down | arp { on | off } |
                    dynamic { on | off } |
                    multicast { on | off } | txqueuelen PACKETS |
                    name NEWNAME |
                    address LLADDR | broadcast LLADDR |
                    mtu MTU }
       ip link show [ DEVICE ]
```

Veja também a página de manual `ip(8)`.

10.2.3 Configurando uma interface Wi-Fi

O programa `iwconfig`, que vem no pacote `wireless-tools`, é usado para interfaces Wi-Fi, juntamente com o `ifconfig` ou `ip`.

Veja a página de manual `iwconfig(8)`.

10.2.4 Configurando uma interface PPP

Se você acessa a Internet através de um modem conectado a uma linha telefônica discada, então a conexão é negociada usando o Protocolo Ponto-a-Ponto (PPP). Tais conexões são acessadas como uma interface de rede `ppp0`, `ppp1` e assim por diante.

Uma interface PPP é gerenciada pelo daemon PPP `pppd` quem vem no pacote `ppp`. Assim, para o usuário, configurar uma interface PPP significa configurar o `pppd`.

Configurando o `pppd` manualmente

Para um link de rede ser estabelecido, precisa-se abrir uma porta de comunicação (normalmente uma porta serial), enviar comandos a um dispositivo de comunicação (normalmente um modem), discar um número de telefone, autenticar a identidade em um daemon PPP remoto, criar uma interface PPP e então modificar tabelas de roteamento de forma que o tráfego possa ser enviado através do link. O `pppd` pode fazer tudo isso e conseqüentemente tem uma longa lista de opções de operação. Essas opções são descritas na página de manual `pppd(8)`.

Em um sistema Debian, as opções globais são definidas no arquivo `/etc/ppp/options`. As opções específicas de usuário são definidas em `~/.ppprc`. Opções que precisam depender da porta de comunicação usada são gravadas em `/etc/ppp/options.`*`nome_da_porta`*. Por exemplo, suponha que você tem dois modems—um modem Lucent LT interno ("on board") acessado através de /dev/LT-modem e um modem externo acessado através de /dev/ttyS0. Crie os dois seguintes arquivos de opções.

```
# cat > /etc/ppp/options.LT-modem <<EOF
115200
init "/usr/sbin/chat -f /etc/chatscripts/setup-LT-modem"
EOF
# cat > /etc/ppp/options.ttyS0 <<EOF
115200
init "/usr/sbin/chat -f /etc/chatscripts/setup-ttyS0"
EOF
```

Esses se referem aos seguites scripts chat. Primeiro, `/etc/chatscripts/setup-LT-modem`.

```
ABORT ERROR
'' ATZ
OK 'ATW2X2 S7=70 S11=55'
OK AT
```

Segundo, `/etc/chatscripts/setup-ttyS0`.

```
ABORT ERROR
```

```
'' ATZ
OK 'ATL1M1Q0V1W2X4&C1&D2 S6=4 S7=70 S11=55 S95=63 S109=1 +FCLASS=0'
OK AT
```

Os conteúdos desses arquivos devem depender de seu hardware, claro.

As opções também podem ser dadas ao `pppd` como argumentos.

No Debian, o `pppd` normalmente é iniciado usando o comando `pon`. Quando o `pon` é usado, seu primeiro argumento dá nome a um arquivo de opções em `/etc/ppp/peers/` que também é lido pelo `pppd`. [5] Esse é o local onde se define as opções que são específicas a um parceiro (peer) particular—por exemplo, um Provedor de Serviço de Internet (ISP) particular.

Suponha por exemplo que você alterne entre Amsterdam e Den Haag. Em cada cidade você tem acesso a dois serviços de ISP—Planet e KPN. Primeiro crie um arquivo de opções básicas para cada ISP.

```
# cat > /etc/ppp/peers/KPN <<EOF
remotename KPN
noauth
user kpn
noipdefault
ipparam KPN
EOF
# cat > /etc/ppp/peers/Planet <<EOF
remotename Planet
auth
user user3579@planet.nl
noipdefault
mru 1000
mtu 1000
ipparam Planet
EOF
```

Esses arquivos definem opções que diferem entre os dois ISPs. Opções comuns a ambos os ISPs podem ser colocadas em `/etc/ppp/options` ou em um dos arquivos de opções específicos de interface, conforme apropriado.

Agora crie arquivos de opções para cada ISP em cada cidade. Em nosso exemplo a única diferença entre conectar a um ISP em um local ou em outro é o chatscript que é necessário. (O chatscript é diferente porque o número de telefone de acesso local é diferente.)

```
# cat > /etc/ppp/peers/KPN-Amsterdam <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/KPN-Amsterdam"
file /etc/ppp/peers/KPN
```

---

[5] Esse arquivo de opções é incluído usando a opção `call`.

```
EOF
# cat > /etc/ppp/peers/KPN-DenHaag <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/KPN-DenHaag"
file /etc/ppp/peers/KPN
EOF
# cat > /etc/ppp/peers/Planet-Amsterdam <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/Planet-Amsterdam"
file /etc/ppp/peers/Planet
EOF
# cat > /etc/ppp/peers/Planet-DenHaag <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/Planet-DenHaag"
file /etc/ppp/peers/Planet
EOF
```

Cada diretiva `file` inclui um dos arquivos de opções mostrados anteriormente. A diretiva `connect` especifica o comando que o `pppd` usa para fazer a conexão. Normalmente usa-se o programa `chat` para isso, adaptando o chatscript para o ISP. Aqui estão os chatscripts para Den Haag; os chatscripts para Amsterdam devem ser similares exceto pelo número de telefone, ou podem ser diferentes se o ISP oferecer serviço através de outra companhia lá.

```
# cat > /etc/chatscripts/KPN-DenHaag <<EOF
ABORT BUSY
ABORT ’NO CARRIER’
ABORT VOICE
ABORT ’NO DIALTONE’
ABORT ’NO DIAL TONE’
ABORT ’NO ANSWER’
ABORT ERROR
OK-AT-OK ATDT 0676012321
CONNECT \d\c
EOF
# cat > /etc/chatscripts/Planet-DenHaag <<EOF
ABORT BUSY
ABORT ’NO CARRIER’
ABORT VOICE
ABORT ’NO DIALTONE’
ABORT ’NO DIAL TONE’
ABORT ’NO ANSWER’
ABORT ERROR
OK-AT-OK ATDT 0676002505
CONNECT \d\c
EOF
```

Para poder conectar a esses ISPs você precisa de nomes de clientes e senhas que o `pppd` possa fornecer ao parceiro (peer) sob demanda. Essa informação é guardada em `/etc/ppp`

`/pap-secrets` (se o protocolo PAP for usado) ou em `/etc/ppp/chap-secrets` (se o protocolo CHAP for usado). Apesar de CHAP ser mais seguro, PAP ainda é o mais largamente usado. Devido a esses arquivos conterem segredos, o grupo e o mundo não podem ter permissão de leitura ou escrita neles. Os formatos desses arquivos são explicados na página de manual `pppd(8)`. Um "segredo" (terceiro campo) é procurado no arquivo encontrando o nome do cliente (primeiro campo) e/ou o nome do servidor (segundo campo). Ao conectar a um ISP geralmente não se sabe o nome do servidor, então ao invés fornece-se o nome do cliente; isso foi feito nas linhas `user` nos `peers/KPN` e `peers/Planet` acima.

```
# nome cliente       nome serv.   segredo
kpn                  *            kpn
user3579@planet.nl   *            myfavoritepet
```

Configurando o `pppd` usando o `pppconfig`

Uma maneira rápida de configurar o `pppd` é usar o programa `pppconfig` que vem no pacote de mesmo nome. O `pppconfig` define arquivos como aqueles acima depois de perguntar ao usuário questões através de uma interface de menu.

Se você optar usar o `resolvconf` então certifique-se de selecionar "None" na tela "Configure Nameservers".

Configurando uma interface PPP usando `wvdial`

Uma abordagem diferente de usar o `pppd` é executá-lo a partir do `wvdial` que vem no pacote `wvdial`. Ao invés de o `pppd` executar o `chat` para discar e negociar a conexão, o `wvdial` faz a discagem e negociação inicial e então inicia o `pppd` para que faça o resto. Dados somente um número de telefone, nome de usuário e senha, o `wvdial` em muitos casos tem sucesso em fazer a conexão.

10.3 Nomeando o computador

10.3.1 Nome do sistema (Hostname)

Um sistema Debian algumas vezes precisa identificar a si mesmo pelo nome. Para esse propósito é mantido um **nome de sistema** (**hostname**) pelo kernel.

O script de inicialização `/etc/init.d/hostname.sh` ajusta o nome do sistema durante a inicialização (usando o comando `hostname`) para o nome gravado em `/etc/hostname`. Esse arquivo contém **somente** o nome do sistema, não um nome de domínio completo.

Para mostrar o nome do sistema atual execute `hostname` sem argumentos.

10.3.2 Nome de correio (Mailname)

O **nome de correio** (mailname) de um sistema é o nome que os programas relacionados a correio usam para identificar o sistema. O arquivo `/etc/mailname` contém esse nome seguido por uma nova linha. O nome de correio geralmente é um dos nomes de domínio totalmente qualificados do sistema. Veja a página de manual `mailname(5)`.

O que o destinatário de e-mail vê no cabeçalho `From:` (De:) de uma mensagem enviada por seu sistema Debian depende de como Agentes de Correio de Usuário (MUA) e Agentes de Transferência de Correio (MTA) são configurados. Suponha que um usuário local *`foo`* envie uma mensagem a partir de um sistema com nome de correio *`meusistema.dom`*. O cabeçalho `From:` do e-mail de saída será:

- "`From:` *`foo@meusistema.dom`*" se o MUA não tiver um cabeçalho `From:` definido;
- "`From:` *`bar@meusistema.dom`*" se o MUA tiver "`From:` *`bar`*" definido;
- "`From:` *`bar@bogus.dom`*" se o MUA tiver "`From:` *`bar@bogus.dom`*" definido.

Mesmo quando o MUA tiver um cabeçalho `From:` definido o MTA pode adicionar um cabeçalho "`Sender:`*`foo@herman.dom`*" para indicar sua verdadeira origem.

É claro que quando algum MTA envolvido realiza reescrita de endereço como discutido em 'Definindo um pega-tudo (catchall) para endereços de email inexistentes sob o Exim' on page 168 e 'Configurando reescrita seletiva de endereço para mensagens de saída sob o Exim' on page 169, o endereço de e-mail visto pelo destinatário pode ser mudado para qualquer coisa.

10.4 Domain Name Service (DNS) (Serviço de Nome de Domínio)

Os sistemas são referenciados por um nome de domínio assim como por endereço IP. DNS é um sistema cliente-servidor em que resolvedores de nome consultam servidores de nome (nameservers) para associar nomes de domínios com endereços IP e outras propriedades de sistemas. O `resolver(3)` da Biblioteca C GNU também pode procurar endereços IP em arquivos ou consultar Network Information Services (NIS) (Serviços de Informação de Rede).

Para ver qual nome de domínio é associado com o sistema local, use o comando `hostname --fqdn`. Isso mostra o primeiro nome de domínio completamente qualificado que o resolvedor encontra para o nome do sistema local. [6]

10.4.1 O resolvedor (resolver)

O trabalho de encontrar quais endereços IP são associados com um nome de domínio particular é trabalho de um resolvedor. O resolvedor mais comumente usado é o conjunto de funções que leva esse nome (`resolver(3)`) na Biblioteca C GNU. Outro é o resolvedor FireDNS que vem no pacote `libfiredns`.

---

[6] Tecnicamente, é o FQDN retornado pelo `gethostbyname(2)` para o nome de sistema retornado pelo `gethostname(2)`.

A forma como o resolvedor da LIBC resolve nomes é governada pela linha `hosts` no arquivo de configuração `/etc/nsswitch.conf`. Essa linha lista os serviços que devem ser usados para resolver um nome: por exemplo, `dns`, `files`, `nis`, `nisplus`. [7] Veja a página de manual `nsswitch.conf(5)`. Se for usado o serviço `files`, o comportamento do resolvedor também é governado pelo arquivo de configuração `/etc/hosts`. Veja a página de manual `hosts(5)`.

Todos os arquivos acima são estáticos e podem ser editados com seu editor favorito.

Se for usado o serviço `dns`, o comportamento do resolvedor também é governado pelo arquivo de configuração `/etc/resolv.conf`. Veja a página de manual `resolv.conf(5)`. Uma das funções importantes do `resolv.conf` é listar os endereços IP de servidores de nome que serão contactados para resolver o nome. Essa lista freqüentemente depende do ambiente de rede e esse pode mudar de tempos em tempos enquanto seu computador está funcionando. Programas como o `pppd` e o `dhclient` são capazes de manipular o `resolv.conf` para adicionar e remover linhas, mas esses recursos não funcionam sempre apropriadamente e eles conflitam uns com os outros. O pacote `resolvconf` soluciona o problema melhor fornecendo uma estrutura padrão para atualizar esse arquivo. Veja 'Gerenciando informação do resolvedor de nomes – `resolvconf`' on the current page.

10.4.2 Gerenciando informação do resolvedor de nomes – `resolvconf`

O pacote `resolvconf` fornece uma estrutura para gerenciamento dinâmico de informação disponível sobre servidores de nomes. Ele soluciona o problema de longa data de como manter listas dinâmicas de servidores de nomes para o resolvedor e caches de DNS para usar. O resolvconf ajusta a si mesmo como o intermediário entre programas que controlam interfaces de rede e fornecem informação de servidor de nome, e aplicações que precisam de informação de servidor de nome.

O `resolvconf` é projetado para funcionar sem que qualquer configuração manual precise ser feita. Entretanto, o pacote é um tanto novo e requer alguma intervenção manual para que funcione adequadamente. Isso é certamente verdadeiro se você já personalizou pacotes para que atualizem o `/etc/resolv.conf`: você precisará desabilitar suas personalizações.

Veja `/usr/share/doc/resolvconf/README.gz` para detalhes.

10.4.3 Colocando em cache nomes consultados – `dnsmasq`

A menos que seu servidor de nomes precise ser obrigatório para um domínio, é melhor que você use um servidor de nomes local para cache como o `dnsmasq`. Ele funciona bem com o `resolvconf`.

---

[7] Diz-se que a forma como o resolvedor resolve nomes também é governada pelo arquivo de configuração `/etc /host.conf`. A linha `order` nesse arquivo lista os métodos que devem ser usados para resolver um nome: por exemplo, `bind`, `hosts`, `nis`. Veja a página de manual `host.conf(5)`. Eu acredito que essa linha foi superada pela linha `hosts` no `nsswitch.conf`, mas não tenho certeza.

10.4.4 Fornecendo Serviço de Nome de Domínio – `bind`

Se você precisa fornecer serviço de nome de autoridade para um domínio então precisa de um servidor de nomes bastante maduro como o `named` que vem no pacote `bind9` ou no pacote `bind`. Recomenda-se o `bind9` para instalações novas.

Para instalar o `bind9`, instale esses pacotes básicos: `bind9`; `dnsutils`. Você pode também desejar instalar esses pacotes utilitários: `bind9-host`; `dns-browse`; `dnscvsutil`; `nslint`. Você também pode querer instalar esse pacote de documentação: `bind9-doc`. Você pode também querer instalar esses pacotes de desenvolvimento: `libbind-dev`; `libnet-dns-perl`. Se você configurar interfaces usando DHCP então pode achar esse pacote útil: `dhcp-dns`.

Instale o `bind9` ou reconfigure-o com `dpkg-reconfigure` para fazer a configuração básica. A configuração consiste na edição do arquivo `named.conf`. No Debian esse arquivo é encontrado em `/etc/bind/` e é usado principalmente para definir as zonas DNS básicas; ele `inclui` dois outros arquivos: `named.conf.local`, usado para definir zonas locais, e `named.conf.options`, usado para definir opções. (O último é processado pelo `resolvconf` para produzir `/var/run/bind/named.options` que é igual ao original exceto que a especificação `forwarders` é uma liista de servidores de nome não locais disponíveis no momento. Para fazer uso disso, mude a linha `include` no `named.conf` de forma que inclua `/var/run/bind/named.options`. Veja 'Gerenciando informação do resolvedor de nomes – `resolvconf`' on the preceding page.)

Os arquivos de base de dados nomeados em `named.conf*` sem caminho completo para o arquivo serão gravados em `/var/cache/bind/`. Esse é o lugar certo para gravar arquivos gerados pelo `named`: por exemplo, arquivos de base de dados para zonas para as quais o daemon é secundário. Arquivos de base de dados estáticos em `/etc/bind/` são e precisam ser referenciados em `named.conf` pelos caminhos completos dos nomes de arquivo. Veja `/usr/share/doc/bind9/README.Debian.gz` para detalhes.

10.5 Configurando interfaces de rede usando DHCP

A configuração de baixo nível de interfaces de rede pode ser automatizada por meio de Dynamic Host Configuration Protocol (DHCP) (Protocolo de Configuração Dinâmica de Sistemas). Seu firewall ou sistema roteador ou seu ISP de banda larga podem fornecer endereços IP e outros parâmetros dessa maneira.

Para fazer isso funcionar você precisa instalar um dos seguintes pacotes:

- `dhcp3-client` (versão 3, Internet Software Consortium)
- `dhcpcd` (Yoichi Hariguchi e Sergei Viznyuk)
- `pump` (Redhat)

O `pump` é simples e largamente utilizado. O `dhcp3-client` é complexo mas mais configurável. [8]

---

[8] Desde outubro de 2003 há também um pacote `dhcp-client` disponível. Ele contém a versão 2 do Cliente DHCP do ISC que foi ultrapassado pela versão 3 que vem no pacote `dhcp3-client`. Os mantenedores planejam mudar o nome de `dhcp3-client` para `dhcp-client` algum dia mas parece que isso não acontecerá a tempo para o lançamento do Sarge.

10.6 Configuração de rede de alto nível no Debian

Para tornar a configuração de rede mais fácil, o Debian fornece uma ferramenta de configuração de rede de alto nível padrão que consiste dos programas `ifup` e `ifdown` e do arquivo `/etc/network/interfaces`. [9] Se você optar por usar `ifupdown` para fazer a configuração de sua rede então normalmente você **não** deve usar comandos de baixo nível também. [10] O `ifupdown` é escrito considerando que será usado sozinho para configurar e desconfigurar interfaces de rede.

Para atualizar configuração de interface faça isso:

```
# ifdown eth0
# editor /etc/network/interfaces  # altere como quiser
# ifup eth0
```

Para mais informações veja `interfaces(5)`, `/usr/share/doc/ifupdown/examples/network-interfaces.gz` e `ifup(8)`.

10.6.1 Configurando uma interface com um endereço IP estático

Suponha que você quer configurar uma interface Ethernet que tem um endereço IP fixo `192.168.0.123`. Esse endereço começa com `192.168.0` então deve estar em uma LAN. Suponha também que `192.168.0.1` é o endereço do gateway da LAN para a Internet. Edite o arquivo `/etc/network/interfaces` de forma que inclua uma estrofe como essa:

```
iface eth0 inet static
        address 192.168.0.123
        netmask 255.255.255.0
        gateway 192.168.0.1
```

Se você tiver o `resolvconf` instalado então pode adicionar linhas que especifiquem informação de DNS. Por exemplo:

```
iface eth0 inet static
        address 192.168.0.123
        netmask 255.255.255.0
        gateway 192.168.0.1
        dns-search nicedomain.org
        dns-nameservers 195.238.2.21 195.238.2.22
```

---

[9] O formato de arquivo do `/etc/network/interfaces` para as versões atuais do `ifupdown` são ligeiramente incompatíveis com o formato de arquivo para as versões do pacote anteriores ao Potato. O script pós-instalação do `ifupdown` deve atualizar o arquivo automaticamente se necessário.

[10] Isso significa também que você não deve usar outras ferramentas de configuração de alto nível como `whereami` para chamar as ferramentas de configuração de baixo nível.

Depois que a interface foi levantada, os argumentos das opções `dns-search` e `dns-nameservers` se tornam disponíveis para o `resolvconf` para inclusão no arquivo `resolv.conf`. O argumento `nicedomain.org` da opção `dns-search` corresponde ao argumento de uma opção `search` em `resolv.conf(5)`. Os argumentos `195.238.2.21` e `195.238.2.22` da opção `dns-nameservers` corresponde aos argumentos das opções `nameserver` no `resolv.conf(5)`. Outras palavras de opção reconhecidas são `dns-domain` e `dns-sortlist`. Veja 'Gerenciando informação do resolvedor de nomes – `resolvconf`' on page 193.

10.6.2 Configurando uma interface usando DHCP

Para configurar uma interface usando DHCP edite o arquivo `/etc/network/interfaces` de forma que inclua uma estrofe como essa:

```
iface eth0 inet dhcp
```

Para que isso funcione você precisa ter instalado um dos clientes DHCP mencionados em 'Configurando interfaces de rede usando DHCP' on page 194.

10.6.3 Configurando uma interface Wi-Fi

O pacote `wireless-tools` inclui um script `/etc/network/if-pre-up.d/wireless-tools` que torna possível configurar hardware Wi-Fi (802.11a/b/g) antes de levantar a interface. A configuração é feita usando o programa `iwconfig`; veja a página de manual `iwconfig(8)`. Para cada parâmetro de comando possível do `iwconfig` você pode incluir uma opção em `/etc/network/interfaces` nomeada como o parâmetro com um prefixo "wireless-". Por exemplo, para definir o ESSID de `eth0` para `meuessid` e a chave de encriptação para `123456789e` antes de levantar a `eth0` usando DHCP, edite o arquivo `/etc/network/interfaces` de forma a incluir uma estrofe como esta:

```
iface eth0 inet dhcp
        wireless-essid myessid
        wireless-key 123456789e
```

10.6.4 Configurando uma interface PPP

Os programas `ifup` e `ifdown` usam `pon` e `poff` para adicionar e remover interfaces PPP, então primeiro leia 'Configurando uma interface PPP' on page 188.

Suponha que você precisa configurar PPP para funcionar com o parceiro (peer) `meuisp`. Edite o `/etc/network/interfaces` de forma a incluir uma estrofe como essa:

```
iface ppp0 inet ppp
        provider myisp
```

Com essa estrofe no lugar, `ifup ppp0` faz

```
pon meuisp
```

Infelizmente atualmente não é possível fornecer opções do `pppd` adicionais em uma estrofe `ppp` no `/etc/network/interfaces`. [11]

Atualmente não é possível usar `ifupdown` para realizar configuração auxiliar de interfaces PPP. Devido ao `pon` finalizar antes que o `pppd` tenha acabado de fazer a conexão, o `ifup` executa os scripts `up` antes que a interface PPP esteja pronta para usar. Até que esse bug [12] seja corrigido é necessário fazer configuração auxiliar em `/etc/ppp/ip-up` ou `/etc/ppp/ip-up.d`.

10.6.5 Configurando uma interface PPPoE

Muitos Internet Service Providers (ISPs) (Provedores de Serviço de Internet) usam PPP para negociar conexões mesmo quando as máquinas dos clientes são conectadas a eles através de redes Ethernet e/ou ATM. Isso é realizado por meio de PPP sobre Ethernet (PPPoE) que é uma técnica para o encapsulamento de fluxos PPP dentro de frames Ethernet. Suponha que seu ISP se chame *`meuisp`*. Primeiro configure PPP e PPPoE para o parceiro (peer) *`meuisp`*. A maneira mais fácil de fazer isso é instalar o pacote `pppoeconf` e executar o `pppoeconf` a partir do console. Então edite o arquivo `/etc/network/interfaces` de forma a incluir uma estrofe como essa:

```
iface eth0 inet ppp
        provider myisp
```

Algumas vezes ocorrem problemas de Maximum Transmit Unit (MTU) (Unidade de Transmissão Máxima) com PPPoE sobre Digital Subscriber Line (DSL). Veja DSL-HOWTO (`http://www.tldp.org/HOWTO/DSL-HOWTO/`) para detalhes.

Note que se seu modem de banda larga contém um roteador então o próprio modem/roteador negocia a conexão PPPoE e aparece no lado da LAN como um simples gateway Ethernet para a Internet.

10.6.6 Configurando múltiplas interfaces Ethernet para um gateway

Suponha que a `eth0` esteja conectada à Internet com um endereço IP configurado por DHCP e que a `eth1` esteja conectada à LAN com um endereço IP estático `192.168.1.1`. Edite o arquivo `/etc/network/interfaces` de forma a incluir estrofes como essas:

---

[11] Veja o bug #196877 (`http://bugs.debian.org/196877`).

[12] Veja o bug #127786 (`http://bugs.debian.org/127786`).

```
iface eth0 inet dhcp

iface eth1 inet static
        address 192.168.1.1
        netmask 255.255.255.0
```

Se você ativar NAT nesse sistema como descrito em 'Construindo um roteador gateway' on page 207, então você poderá compartilhar a conexão à Internet com todos os sistemas na LAN.

10.6.7 Configurando interfaces virtuais

Usando interfaces virtuais você pode configurar uma única placa Ethernet para ser uma interface para várias sub-redes IP. Por exemplo, suponha que seu sistema esteja em uma rede LAN 192.168.0.x/24. Você quer conectar o sistema à Internet usando um endereço IP público provido via DHCP usando sua placa Ethernet existente. Edite o `/etc/network/interfaces` de forma a incluir estrofes como essas:

```
iface eth0 inet static
        address 192.168.0.1
        netmask 255.255.255.0
        network 192.168.0.0
        broadcast 192.168.0.255

iface eth0:0 inet dhcp
```

A interface `eth0:0` é uma interface virtual. Quando é levantada, também é levantada a sua superior `eth0`.

10.7 Reconfiguração de rede

O que vem a seguir será importante para o leitor compreender a diferença entre uma **interface física** e uma **interface lógica**. [13] Uma interface física é o que temos chamado de "a interface", a coisa que é chamada `eth0`, `ppp1`, ou o que você tiver. Uma interface lógica é um conjunto de valores que pode ser atribuído aos parâmetros variáveis de uma interface física. Se você achar isso confuso, substitua a expressão "configurado como interface lógica `X`" pela expressão "configurado com o perfil de interface `X`" conforme você ler.

As definições `iface` no arquivo `/etc/network/interfaces` são na verdade definições de interfaces lógicas, não de interfaces físicas. [14] Se você nunca quiser reconfigurar suas interfaces então pode ignorar esse fato já que a interface física `X` por padrão será configurada como interface lógica `X`.

---

[13] Essa terminologia é usada na documentação do `ifupdown`.

[14] Note que as interfaces nas linhas `auto` são interfaces físicas, não interfaces lógicas.

Entretanto, suponha que seu computador seja um laptop que você transporta entre a casa e o trabalho. Quando você conecta o computador à rede corporativa ou à sua LAN doméstica você precisa configurar a `eth0` apropriadamente.

Primeiro defina duas interfaces lógicas `casa` e `trabalho` (ao invés de `eth0` como fizemos anteriormente) que descrevem como a interface deve ser configurada para a rede doméstica e a rede do trabalho, respectivamente.

```
iface home inet static
        address 192.168.0.123
        netmask 255.255.255.0
        gateway 192.168.0.1

iface work inet static
        address 81.201.3.123
        netmask 255.255.0.0
        gateway 81.201.1.1
```

Então a interface física `eth0` pode ser levantada para a rede doméstica com a configuração apropriada especificando na linha de comando:

```
# ifup eth0=casa
```

Para configurar a `eth0` para a rede do trabalho execute os comandos:

```
# ifdown eth0
# ifup eth0=work
```

Note que com o arquivo `interfaces` escrito como acima não será mais possível levantar a `eth0` usando apenas `ifup eth0`. A razão para isso é que o `ifup` usa o nome da interface física como o nome padrão da interface lógica e agora em nosso exemplo não há uma interface lógica `eth0` definida.

10.8 Reconfiguração mágica de rede

Nomes de interface lógica podem ser ”mapeados“ para outros nomes de interface lógica quando o `ifup` é executado. A forma como os nomes são mapeados pode ser feita para depender de circunstâncias. Então o `ifup` pode ser configurado de forma que levante uma dada interface física como a interface lógica apropriada entre um conjunto de alternativas predefinidas.

O mapeamento de nome de interface lógica ocorre como a seguir:

- Se nenhum nome de interface lógica for dado na linha de comando do `ifup` então o nome da interface física é usado como o nome de interface lógica inicial.

- Se o nome de interface lógica combinar com o padrão-glob de uma estrofe `mapping` então o mapeamento é aplicado para gerar um novo nome de interface lógica. Isso é feito recursivamente até não haver padrões-glob que combinem.
- Se o último nome de interface lógica for o rótulo de uma definição de interface lógica em `/etc/network/interfaces` então a interface física é levantada como essa interface lógica. Caso contrário o `ifup` mostra uma mensagem que está ignorando a interface desconhecida ("Ignoring unknown interface" e sai.

A sintaxe de uma estrofe `mapping` é:

```
mapping padrão-glob
        script nome-do-script
        [map entrada do script]
```

O script nomeado na estrofe `mapping` é executado com o nome de interface física como seu argumento e com os conteúdos de todas as linhas "map" seguintes na estrofe para ele como sua entrada padrão. O script mostra o resultado do mapeamento em sua saída padrão antes de acabar.

Por exemplo, a seguinte estrofe `mapping` fará com que o `ifup` levante a interface `eth0` como a interface lógica `casa`.

```
mapping eth0
        script /usr/local/sbin/echo-casa
```

onde `/usr/local/sbin/echo-casa` é:

```
#!/bin/sh
echo casa
```

Devido ao mapeamento ser feito com um script é possível selecionar a interface lógica baseado em algum tipo de teste. Vamos começar com um exemplo. Suponha que você tem placas de rede diferentes para casa e trabalho. O diretório `/usr/share/doc/ifupdown/examples/` contém um script de mapeamento que pode ser usado para selecionar uma interface lógica baseado no endereço de Media Access Controller (endereço MAC) (Controlador de Acesso de Mídia). Primeiro instale o script no diretório apropriado.

```
# install -m770 /usr/share/doc/ifupdown/examples/get-mac-address.sh \
   /usr/local/sbin/
```

Então adicione uma estrofe como a seguinte em `/etc/network/interfaces`:

```
mapping eth0
        script /usr/local/sbin/get-mac-address.sh
        map 02:23:45:3C:45:3C casa
        map 00:A3:03:63:26:93 trabalho
```

Programas de mapeamento mais sofisticados são descritos abaixo em 'guessnet' on the next page, e 'laptop-net' on the facing page.

10.8.1 guessnet

Instale o pacote `guessnet` e então adicione uma estrofe como a seguinte ao arquivo `/etc/network/interfaces`:

```
mapping eth0
        script guessnet-ifupdown
        map casa
        map trabalho
```

Agora quando você fizer `ifup eth0`, o `guessnet` verificará se a `eth0` deve ser levantada como `casa` ou `trabalho`. Para fazer isso ele usa informação gravada nas definições de interface lógica.

10.8.2 laptop-net

O pacote `laptop-net` toma uma abordagem diferente para reconfiguração de rede automágica. O laptop-net não faz uso das interfaces lógicas do `ifupdown` mas ao invés disso tem seu próprio sistema de "esquemas" de configuração e "perfis" de sistema. O laptop-net entretanto ainda usa o `ifupdown` para configurar interfaces físicas. Para mais informações consulte a documentação bem escrita no pacote `laptop-net-doc`.

10.9 Gatilhando a reconfiguração de rede

Nós temos visto como as interfaces podem ser reconfiguradas. A reconfiguração precisa ser feita em momentos apropriados.

Tradicionalmente a rede era configurada durante a seqüência de inicialização através do script de inicialização `/etc/rcS.d/S40networking` e era raramente reconfigurada. Serviços que dependiam de rede eram iniciados mais tarde na seqüência de inicialização. No desligamento ou reinicialização os scripts de inicialização eram executados na ordem oposta.

Atualmente, entretanto, há uma tendência no GNU e no Linux no sentido de suportar hardware e circunstâncias que mudam dinamicamente. O primeiro suporte foi adicionado para cartões PCMCIA trocáveis durante funcionamento; mais recentemente o mecanismo `hotplug` foi adicionado de forma que muito mais periféricos podem ser inseridos e retirados enquanto o computador está funcionando. Isso inclui hardware de rede. Note que serviços que dependem de hardware que é inserido com computador em funcionamento (hot swapped) precisam ser iniciados somente depois que o hardware é inserido e precisam ser parados quando o hardware estiver removido. Isso significa que tais serviços precisam ser removidos do controle de inicialização System V e ao invés disso colocados no controle do `ifupdown`.

Por exemplo, suponha que o serviço `foo` controlado pelo script de inicialização `/etc/init.d/foo` depende da interface de rede `eth0` configurada dinamicamente.

- Primeiro remova `foo` do controle do sistema de inicialização:

```
# update-rc.d -f foo remove
```

- Então coloque `foo` sob controle do `ifupdown` adicionando as opções `up` e `down` à estrofe `eth0` em `/etc/network/interfaces` que contém chamadas ao script de inicialização `foo`:

```
iface eth0 inet dhcp
        up /etc/init.d/foo start
        down /etc/init.d/foo stop
```

10.9.1 Gatilhando a configuração de rede durante a inicialização

Na inicialização o script `/etc/rcS.d/S40networking` executa o comando `ifup -a`. Isso levanta todas as interfaces físicas listadas em estrofes `auto` no `/etc/network/interfaces`.

Atualmente é freqüentemente melhor gerenciar a configuração de rede usando métodos dinâmicos. Uma vez que estão disponíveis mecanismos para suportar hardware que mude dinamicamente se torna mais simples tratar hardware estático como se fosse dinâmico também. A inicialização pode então ser tratada simplesmente como outro evento hotplug. (Veja 'Gatilhando configuração de rede – `hotplug`' on the current page.)

Entretanto, na maioria dos casos se deseja que pelo menos a interface de loopback `lo` seja levantada na inicialização. Assim, certifique-se de que o `/etc/network/interfaces` inclua as seguintes estrofes.

```
auto lo

iface lo inet loopback
```

Você pode listar nomes de interfaces físicas adicionais em estrofes `auto` se desejar que também sejam levantadas durante a inicialização. **Nunca** liste interfaces PCMCIA em estrofes `auto`. O programa `cardmgr` de PCMCIA é iniciado mais tarde na seqüência de inicialização que quando o `/etc/rcS.d/S40networking` é executado.

10.9.2 Gatilhando configuração de rede – `hotplug`

Para obter suporte a hot plug instale o pacote `hotplug`.

Hardware de rede pode ser conectado com o computador ligado durante a inicialização ou depois que um cartão (por exemplo, um cartão PCMCIA) é inserido na máquina ou depois que um utilitário como o `discover` é executado e carrega os módulos de controladores necessários.

Quando o kernel detecta um novo hardware ele inicializa o controlador (driver) para o hardware e então executa o programa `hotplug` para configurá-lo. Mais tarde, se o hardware for removido, então o kernel executa o `hotplug` novamente com ajustes de variáveis de ambiente diferentes. No Debian, quando o `hotplug` é chamado ele executa scripts em `/etc/hotplug/` e `/etc/hotplug.d/`. Veja `hotplug(8)` para obter detalhes.

O hardware de rede recentemente inserido é configurado pelo script `/etc/hotplug/net.agent`. [15] Suponha que seu cartão de rede PCMCIA tenha sido inserido resultando na interface `eth0` se tornando disponível para uso. O `/etc/hotplug/net.agent` faz o seguinte:

```
ifup eth0=hotplug
```

A menos que você tenha adicionado uma definição de interface lógica ou um mapeamento chamado `hotplug` em `/etc/network/interfaces`, esse comando não fará nada. Para fazer com que o comando configure a `eth0`, adicione a seguinte estrofe em `/etc/network/interfaces`:

```
mapping hotplug
        script echo
```

(**Não** inclua uma estrofe mapping como essa se você estiver usando pedidos do `ifplugd` iniciados pelo `hotplug` para controlar a interface, como descrito em 'Gatilhando a configuração de rede – `ifplugd`' on this page.)

Se você quiser que apenas a `eth0` e mais nenhuma outra interface seja levantada com hot plug então use `grep` ao invés de `echo` como a seguir:

```
mapping hotplug
        script grep
        map eth0
```

Veja 'Reconfiguração mágica de rede' on page 199 e `/usr/share/doc/hotplug/README.Debian` para mais dicas.

10.9.3 Gatilhando a configuração de rede – `ifplugd`

O `ifplugd` levanta ou derruba uma interface de acordo com o hardware relacionado estar ou não plugado em uma rede. O programa pode detectar um cabo conectado a uma interface Ethernet ou um ponto de acesso associado a uma interface Wi-Fi. Quando o `ifplugd` vê que o estado do link foi alterado ele executa um script proxy que por padrão chama o `ifup` ou `ifdown` para a interface.

O `ifplugd` funciona bem em combinação com o `hotplug`. Quando um cartão foi inserido resultando na disponibilização de uma interface para uso, `/etc/hotplug.d/net/ifplugd.hotplug` inicia um pedido de `ifplugd` para essa interface. Quando o `ifplugd` detecta que o cartão está plugado em uma rede ele usa o `ifup` para a interface.

---

[15] Ele também pode ser configurado por quaisquer scripts que tenham sido instalados em `/etc/hotplug.d/net/`.

10.9.4 Gatilhando a configuração de rede – `waproamd`

Para associar a um ponto de acesso, um cartão Wi-Fi pode ter que ser programado com uma chave de encriptação WEP apropriada. Se você está usando o `ifplugd` para controlar o `ifup` como descrito em 'Gatilhando a configuração de rede – `ifplugd`' on the preceding page então obviamente você não pode definir a chave de encriptação usando o `ifup` porque o `ifup` só é chamado depois que o cartão foi associado. Uma possível solução é programar todas as chaves requeridas na memória não-volátil do cartão Wi-Fi. Se você muda entre muitas redes então seu cartão Wi-Fi pode não ser capaz de gravar chaves suficientes.

Outra solução é usar o `waproamd` que define a chave de encriptação WEP de acordo com os pontos de acesso que estão disponíveis, conforme revelado pela varredura (scanning).

O `waproamd` funciona bem em combinação com o `hotplug`. Quando um cartão foi inserido resultando na disponibilização de uma interface para uso, o `/etc/hotplug.d/net/waproamd` inicia um pedido de `waproamd` para essa interface. O `waproamd` funciona muito bem em combinação com o `ifplugd`.

Veja o arquivo README do pacote `waproamd` para mais informações.

10.9.5 Configuração de rede e PCMCIA

Se você usa cartões de rede PCMCIA de 16 bit então deve definir `CARDMGR_OPTS="-f"` em `/etc/default/pcmcia`. Isso deixa a inicialização ligeiramente mais lenta mas previne uma condição de corrida executando o `cardmgr` em primeiro plano (foreground) até que todos os cartões PCMCIA de 16 bit sejam configurados.

Apesar de por padrão o `/etc/init.d/pcmcia` ser iniciado a partir de `/etc/rc2.d/S20pcmcia` você pode querer mudar isso para executar antes, por exemplo, para `/etc/rc2.d/S12pcmcia` para garantir que o subsistema PCMCIA seja inicializado antes que os serviços de rede sejam iniciados em S20.

Há várias abordagens possíveis para configurar interfaces de rede PCMCIA.

- Para cartões de rede PCMCIA de 32 bit PCI (CardBus):
  - `hotplug` / `ifupdown`
- Para cartões de rede PCMCIA de 16 bit ISA:
  - `hotplug` / `ifupdown` com `/etc/pcmcia/network` desabilitado localmente (recomendado), ou
  - `pcmcia-cs` / `ifupdown` com o `/etc/pcmcia/network` padrão (obsoleto), ou
  - `pcmcia-cs` personalizado habilitando recursos de `/etc/pcmcia/network` (obsoleto)

A abordagem recomendada para cartões de 16 bit toma vantagem do fato que o subsistema hot plug do Linux 2.4 agora suporta PCMCIA. Simplesmente siga as direções em 'Gatilhando configuração de rede – `hotplug`' on page 202. [16] Note, entretanto, que para prevenir que

---

[16] Em versões anteriores do Debian a forma padrão de configurar cartões de rede PCMCIA era através dos scripts hook de `cardmgr`, `/etc/pcmcia/network` e `/etc/pcmcia/network.opts`. Esses scripts hook foram

o script hook `/etc/pcmcia/network` do `cardmgr` interfira com o `hotplug` você precisa adicionar localmente a linha:

```
exit 0
```

no início de `/etc/pcmcia/network` para desabilitar esse comportamento padrão.

Note que não há nada errado em executar o `cardmgr`. Nós apenas não queremos que ele chame programas de configuração de rede.

Para o `cardmgr` funcionar adequadamente você precisa editar o arquivo `/etc/pcmcia/config.opts` para configurar recursos atribuídos a cartões PCMCIA de 16 bit. Veja 'PCMCIA' on page 104 e o HOWTO de PCMCIA no Linux (`http://www.tldp.org/HOWTO/PCMCIA-HOWTO.html`) para mais informações.

10.10 Configuração de serviço de rede

A configuração de serviço de rede típica em um ambiente desktop ou servidor doméstico envolve:

- O *super-servidor* de Internet e daemon wrapper TCP/IP, veja 'Restringindo acesso a serviços' on page 142.
  - `/etc/inetd.conf`
- `ssh`: Shell segura OpenSSH, veja 'SSH' on page 163.
  - `/etc/ssh/ssh_config`
  - `/etc/ssh/sshd_config`
- `exim`: agente de transporte de correio, veja 'Nome de correio (Mailname)' on page 192 e 'Agentes de transporte de correio (MTAs)' on page 167.
  - `/etc/exim/exim.conf`
  - `/etc/mailname`
  - `/etc/aliases`
  - `/etc/email-addresses`
- `fetchmail`: daemon para baixar mensagens de uma conta POP3, veja 'Baixando mensagens – Fetchmail' on page 170.
  - `/etc/fetchmailrc`
- `procmail`: programa de distribuição local de mensagens e filtro, veja 'Processando mensagens – Procmail' on page 170.
  - `~/.procmailrc`
- Hostname e DNS (proxy, cache, . . .), veja 'Nome do sistema (Hostname)' on page 191 e 'Domain Name Service (DNS) (Serviço de Nome de Domínio)' on page 192.

---

desenvolvidos em época anterior a uma capacitação do Linux para hot plug para propósitos mais gerais. Algumas pessoas ainda usam esses scripts em seu estado padrão quando eles simplesmente chamam `ifup` depois que a interface foi adicionada e `ifdown` quando a interface é removida. Como notado acima, agora é recomendado usar o `hotplug` para fazer isso. Outros ainda usam o sistema especial de chamada de comandos de configuração de rede de baixo nível que são ativados quando certas variáveis em `/etc/pcmcia/network.opts` são definidas para "`y`". Esse sistema tem vários problemas. Ele é afligido por condições de corrida; só funciona para cartões PCMCIA de 16 bit; ele faz o que é melhor deixar para o `ifupdown` fazer. Conseqüentemente está desatualizado.

  - `/etc/host.conf`
  - `/etc/hostname`
  - `/etc/hosts`
  - `/etc/hosts.allow`
  - `/etc/hosts.deny`
  - `/etc/resolv.conf`
  - `/etc/bind/named.conf` (edite)
  - `/etc/bind/db.lan` (adicione para sistemas de LAN)
  - `/etc/bind/db.`*`192.168.0`* (adicione para LAN reverso)
- DHCP, veja 'Configurando interfaces de rede usando DHCP' on page 194.
  - `/etc/dhcp3/dhclient.conf` (DHCP lado do cliente)
  - `/etc/default/dhcp3-server` (DHCP lado do servidor)
  - `/etc/dhcp3/dhcpd.conf` (DHCP lado do servidor)
- `cvs`: concurrent versions system (sistema de versões concorrentes), veja 'Concurrent Versions System (CVS) (Sistema de Versões Concorrentes)' on page 217.
  - `/etc/cvs-cron.conf`
  - `/etc/cvs-pserver.conf`
- `nfs-kernel-server`: network file system (sistema de arquivos de rede), veja 'Configuração de NFS' on page 40. (para sistemas semelhantes ao unix)
  - `/etc/exports`
- `samba`: compartilhamento de arquivos e impressora com o Windows, veja 'Configuração do Samba' on page 40 e 'Samba' on page 134.
  - `/etc/samba/smb.conf`
- Sistema daemon de impressão, veja 'Configuração da Impressora' on page 41.
  - `/etc/printcap` (para o lpr)
- `apache` e `apache2`: servidor web.
  - `/etc/apache/*`
  - `/etc/apache2/*`
- `squid`: servidor cache proxy web.
  - `/etc/squid/*`

10.11 Resolução de problemas com rede

Se você encontrar problemas então verifique a saída do seguinte como primeira verificação do que está acontecendo:

```
# ifconfig
# cat /proc/pci
# cat /proc/interrupts
# dmesg | more
```

Veja também as seções seguindo 'Testes de rede básicos' on page 130.

Se você tiver problemas com alguns sítios web, veja 'Estranhos problemas de acesso a alguns websites' on page 45.

10.12 Construindo um roteador gateway

Um sistema Debian pode ser uma máquina gateway de propósitos gerais que faz Network Address Translation (Translação de Endereço de Rede) (NAT, também conhecido como masquerading (mascaramento)), transferência de correio, DHCP, cache DNS, cache proxy HTTP, serviço CVS, serviço NFS, e serviço Samba. Veja 'Hosts e IP a usar para LAN' on page 31 para o exemplo de tal configuração.

10.12.1 Configuração do Netfilter

O projeto netfilter/iptables é um subsistema de firewalling para o kernel Linux 2.4 e superiores. Consulte Netfilter (`http://www.netfilter.org/`), onde muitos tópicos de configuração de rede são explicados.

Básico do netfilter

O Netfilter processa pacotes usando 5 chains embutidas : PREROUTING, INPUT, FORWARD, OUTPUT e POSTROUTING.

```
                decisão de
                roteamento
IN ------> PRE ---> ------> FORWARD -----> ----> POST -----> OUT
interface  ROUTING  \       filter       /       ROUTING     interface
           DNAT     |       tracking     ^       SNAT
           REDIRECT |                    |       MASQUERADE
                    v                    |
                  INPUT                OUTPUT
                    | filter             ^ filter,DNAT
                    v                    |
                    \--> Processo Local --/
                        programas user-space
```

Tabela Netfilter

Os pacotes são processados em cada chain embutida usando as seguintes tabelas.

- filter (filtro de pacotes, padrão)
    - INPUT (para pacotes entrando na própria máquina)
    - FORWARD (para pacotes sendo roteados através da máquina)
    - OUTPUT (para pacotes gerados localmente).
- nat (tradução de endereços de rede)
    - PREROUTING (para alterar pacotes no momento que eles chegam)
    - OUTPUT (para alterar pacotes gerados localmente antes do roteamento)
    - POSTROUTING (para alterar pacotes quando eles estão prestes a saírem)

- mangle ("mangling" de endereços de rede, bom somente após o kernel 2.4.18)
  - todas as cinco chains embutidas.

Alvo Netfilter

Regras de firewall possuem diversos alvos :

- quatro alvos básicos :
  - ACCEPT significa deixar o pacote passar.
  - DROP significa descartar o pacote.
  - QUEUE significa passar o pacote para o userspace (caso suportado pelo kernel).
  - RETURN significa parar de atravessar esta chain e continuar na próxima regra na chain anterior (a chain que chamou esta).
- alvos extendidos :
  - LOG liga o logging do kernel.
  - REJECT envia como resposta um pacote de erro e descarta o pacote.
  - SNAT altera o endereço de origem de pacote e é usado somente na chain POSTROUTING. (somente tabela nat)

    `--to-source  endereçoip[-endereçoip][:porta-porta]`
  - MASQUERADE é o mesmo que SNAT mas para conexões com endereços IP atribuídos dinamicamente (discadas). (somente tabela nat)

    `--to-ports porta[-porta]`
  - DNAT altera o endereço de destino do pacote e é usado nas chains PREROUTING e OUTPUT e em chains definidas pelo usuário que são chamadas somente a partir destas chains. (somente tabela nat)

    `--to-destination endereçoip[-endereçoip][:porta-porta]`
  - REDIRECT altera o endereço IP de destino para enviar o pacote para a própria máquina.

    `--to-ports porta[-porta]`

Comandos do Netfilter

Os comandos básicos do `iptables` são :

```
iptables -N chain                    # cria uma chain

iptables -A chain \                  # adiciona regra na chain
         -t tabela \                 # usa tabela (filter, nat, mangle)
         -p protocolo \              # tcp, udp, icmp ou all,
         -s endereço-origem[/máscara] \
         --sport porta[:porta] \     # porta de origem caso -p seja tcp ou udp
         -d endereço-origem[/máscara] \
         --dport porta[:porta] \     # porta de destino caso -p seja tcp ou udp
         -j alvo \                   # o que fazer se o pacote casar
         -i nome-interface-entrada \# para INPUT,  FORWARD, PREROUTING
         -o nome-interface-saída     # para FORWARD, OUTPUT, POSTROUTING
```

Network Address Translation (Translação de Endereço de Rede)

Máquinas em uma LAN podem acessar os recursos da Internet através de um gateway que translada o endereço IP na LAN para endereços IP usáveis na Internet.

```
# apt-get install ipmasq
```

Aplique as regras de exemplo para aumentar a proteção do `ipmasq`. Consulte `/usr/share/doc/ipmasq/examples/stronger/README`. Para imagens de kernel 2.4 do Debian sob o woody, certifique-se de carregar os módulos apropriados. A versão do Sarge do ipmasq resolveu esse problema. Consulte 'Função de rede' on page 105 para obter instruções de configuração.

Para imagens de kernel 2.2 do Debian, edite o `Z92timeouts.rul` em `/etc/masq/rules` como a seguir para assegurar uma conexão mais longa para sites remotos (bom para e-mails grandes, etc.) :

```
# tcp, tcp-fin, udp
# 2hr, 10 sec, 160 sec - padrão
# 1 day, 10 min, 10 min - exemplo mais longo
$IPCHAINS -M -S 86400 600 600
```

Adicionalmente, caso a rede seja acessada através de um cartão de rede PCMCIA, o `ipmasq` precisa ser iniciado a partir de `/etc/pcmcia/network.opts` (leia: `/usr/share/doc/ipmasq/ipmasq.txt.gz`) ou a partir de `/etc/network/interfaces` (leia: 'Configuração de rede e PCMCIA' on page 204 e 'Gatilhando a reconfiguração de rede' on page 201).

Redirecionar conexões SMTP (kernel 2.4)

Suponha que você possua um notebook PC que esteja configurado para usar outros ambientes LAN e você queira usar seu agente de mensagens de usuário (software de e-mail) no notebook PC sem reconfigurá-lo.

Adicionar as seguintes regras através do comando `iptables` na máquina gateway irá redirecionar a conexão SMTP para a máquina gateway.

```
# iptables -t nat -A PREROUTING -s 192.168.1.0/24 -j REDIRECT \
           -p tcp --dport smtp --to-port 25 # smtp=25, INPUT está aberto
```

Para um conjunto de regras de redirecionamento mais completo considere instalar o pacote `ipmasq` e adicionar `M30redirect.def` (`http://www.debian.org/doc/manuals/debian-reference/examples/`) ao diretório `/etc/ipmasq/rules/`.

10.12.2 Gerenciar múltiplas conexões de rede

[FIXME] Política de roteamento (por Phil Brutsche `<pbrutsch@tux.creighton.edu>`): Confira o manual iproute (`http://lartc.org/`) para maiores detalhes. Controle de tráfego (tc) pode também ser interessante.

Ambiente :

```
eth0: 192.168.1.2/24; gateway 192.168.1.1
eth1: 10.0.0.2/24; gateway 10.0.0.1
Sem mascaramento nesta máquina.
```

Mágica especial :

```
ip rule add from 192.168.1.2 lookup 1
ip rule add from 10.0.0.2 lookup 2
ip route add to default via 10.0.0.1 metric 0
ip route add to default via 192.168.1.1 metric 1
ip route add table 1 to 192.168.1.0/24 via eth0
ip route add table 1 to 10.0.0.2/24 via eth1
ip route add table 1 to default via 192.168.1.1
ip route add table 2 to 192.168.1.0/24 via eth0
ip route add table 2 to 10.0.0.2/24 via eth1
ip route add table 2 to default via 10.0.0.2
```

[FIXME] Eu nunca fiz isso. Como configurar a conexão discada como backup para uma conexão rápida com recursos de auto-discagem ? Por favor enviem-me um patch aqui :)

Capítulo 11

Editores

11.1 Editores populares

O Linux oferece muitas alternativas para editores de texto de console. Entre eles:

- `vim`: Editor poderoso e leve, herdado do BSD. VI iMproved.
- `emacs`: Editor definitivo e pesado, herdado do GNU. Original de RMS (Richard M. Stallman).
- `xemacs`: Emacs: A Próxima Geração, originalmente de Lucid.
- `mcedit`: Editor GNU para novatos. Idêntico ao editor interno do `mc`. Veja 'Editor no MC' on page 52.
- `ae`: Pequeno editor padrão (Potato). Evite-o.
- `nano`: Pequeno editor GNU padrão (Woody). Emula o `pico`.
- `joe`: Para saudosistas do WordStar ou TurboPascal.
- `jed`: Editor rápido, com muitos recursos e direcionado a menus com atalhos de tecla do Emacs.
- `jove`: Editor muito pequeno com teclas de atalho do emacs.
- `nvi`: Novo vi. Compatível Bug-a-bug com o vi original.

Use `update-alternatives --config editor` para mudar o editor padrão. Além disso, muitos programas usam as variáveis de ambiente `EDITOR` ou `VISUAL` para decidir qual editor usar. Veja 'Editor no MC' on page 52.

Há também alguns editores de texto baseados no X notáveis:

- `gvim`: Vim com GUI (pacote `vim` e pacote `vim-gtk`)
- `emacs`: O Verdadeiro e Único Emacs (auto-detecta X).
- `xemacs`: Próxima geração do Emacs (auto-detecta o X).

Esses comandos xclient usam opções padrões como `-fn a24`, que tornam mais fácil a vida de pessoas mais velhas como eu :) Veja 'Clientes X' on page 151.

11.2 Editores de recuperação

Há uns poucos editores que se localizam em `/bin`. Um desses deve ser instalado para facilitar a edição de arquivos quando `/usr` não estiver acessível.

- `elvis-tiny`: Editor vi mínimo (`vi` para iniciar)
- `nano-tiny`: Editor não-vi mínimo (`nano-tiny` para iniciar)
- `ed`: Editor mínimo (sempre lá mas difícil de usar)

11.3 Emacs e Vim

11.3.1 Dicas para o Vim

Leia o documento ”VIM - main help file“ pressionando <F1> ao executar o programa.

`<F1>`	Ajuda
`<esc>`	Volta ao modo normal
`V`	Modo Visual
`i`	Modo Inserção
`:`	Comandos de linha de comando
`:set tw=72`	Ajusta largura do texto para 72
`<F11>`	Modo insere (cola)
`:r! date -R`	Insere data RFC-822
`q`*a*	Grava teclas de atalho no registrador *a*
`q`	Interrompe a gravação de teclas de atalho
`@`*a*	Executa teclas de atalho do registrador *a*
`:edit` *foo.txt*	Edita outro arquivo carregando *foo.txt*
`:wnext`	Grava o arquivo atual e edita o próximo arquivo

`q` e `@` podem ser usados para registro simples de macro e execução. Por exemplo, para criar macro que insira tags de itálico HTML ao redor de uma palavra no cursor, você poderia entrar `qii<i>^[ea</i>^[q` (onde `^[` é a tecla ESC). Então digitar `@i` no começo de uma palavra adicionaria as tags <i> e </i>.

Consulte também ‘Usando GnuPG com o Vim’ on page 247.

11.3.2 Dicas para o Emacs

`<F1>`	Ajuda
`<F10>`	Menu
`C-u M-! date -R`	Insere data RFC-822

11.3.3 Iniciando o editor

```
iniciar o editor:                emacs filename  vim filename
iniciar em compatível vi:                        vim -C
iniciar em não compatível vi:                    vim -N
iniciar com padrão compilado:    emacs -q        vim -N -u NONE
```

11.3.4 Resumo de comandos do Editor (Emacs, Vim)

```
sair:                            C-x C-c         :qa /:wq /:xa /:q!
voltar/modo comando:             C-g             <esc>
para trás(esquerda):             C-b             h
para Frente(direita):            C-f             l
próxima(abaixo):                 C-n             j
anterior(acima):                 C-p             k
início da linhA(^):              C-a             0
fim da linha($):                 C-e             $
mÚltiplos comandos:              C-u nnn cmd     nnn cmd
Múltiplos comandos:              M-digitkey cmd
salvar arquivo:                  C-x C-s         :w file
início do buffer:                M-<             1G
fim do buffer:                   M->             G
rola para frente 1 tela:         C-v             ^F
rola para frente 1/2 tela:                       ^D
rola para frente 1 linha:                        ^E
rola para trás 1 tela:           M-v             ^B
rola para trás 1/2 tela:                         ^U
rola para trás 1 linha:                          ^Y
rola a outra janela:             M-C-v
apaga sob o cursor:              C-d             x
apaga do cursor ao fim linha:    C-k             D
iSearch para frente:             C-s
isearch para trás:               C-r
procura à frente:                C-s enter       /
procura para tRás:               C-r enter       ?
isearch regexp:                  M-C-s
isearch regexp para trás:        M-C-r
procura regexp:                  M-C-s enter     /
procura regexp para trás:        M-C-r enter     ?
ajuda:                           C-h C-h         :help
ajuda Apropos:                   C-h a
ajuda comandos teclado:          C-h b           :help [key]
ajuda Info:                      C-h i
ajuda Major mode:                C-h m
ajuda Tutorial:                  C-h t           :help howto
```

```
desfazer:                       C-_             u
Refazer:                        C-f             ^R
Marcar posição do cursor:       C-@             m{a-zA-Z}
trocar marca e posição:         C-x C-x
ir para marca arquivo atual:                    '{a-z}
ir para marca qualquer arquivo:                 '{A-Z}
copiar região:                  M-w             {visual}y
recortar região:                C-w             {visual}d
colar e manter buffer:          C-y
colar do buffer recortado:      M-y             p
converte região em maiúsculas:  C-x C-u         {visual}U
converte região em minúsculas:  C-x C-l         {visual}u
Inserir caracter especial:      C-q octalnum/keystroke
                                                ^V decimal/keystroke
substituir:                     M-x replace-string      :%s/aaa/bbb/g
substituir regexp:              M-x replace-regexp      :%s/aaa/bbb/g
consultar e substituir:         M-%                     :%s/aaa/bbb/gc
consultar e substituir:         M-x query-replace
consultar e substituir regexp:  M-x query-replace-regexp
Abrir arquivo:                  C-x C-f         :r file
Salvar arquivo:                 C-x C-s         :w
Salvar todos os buffers:        C-x s           :wa
Salvar como:                    C-x C-w file    :w file
Mudar de buffer:                C-x b
Listar buffers:                 C-x C-b         :buffers
Trocar somente leitura:         C-x C-q         :set ro
Fechar buffer indicado:         C-x k
Dividir na vertical:            C-x 2           :split
Dividir na horizontal:          C-x 3           :vsplit (ver. 6)
Ir para outra janela:           C-x o           ^Wp
Excluir esta janela:            C-x 0           :q
Excluir outra(s) janela(s):     C-x 1           ^Wo
Executar shell em bg:           M-x compile
Destruir shell em bg:           M-x kill-compilation
Executar make:                                  :make Makefile
verificar mensagem de erro:     C-x`            :echo errmsg
executar shell e gravar:        M-x shell       :!script -a tmp
...limpar BS, ...                                :!col -b <tmp >record
...salvar/reler registro shell: C-x C-w record  :r record
executar shell:                 M-! sh          :sh
executar comando                M-! cmd         :!cmd
executar comando e inserir:     C-u M-! cmd     :r!cmd
executar filtro:                M-| file        {visual}:w file
executar filtro e inserir:      C-u M-| filter  {visual}:!filter
exibir opção                                    :se[t] {option}?
redefinir opção para padrão                     :se[t] {option}&
```

```
redefinir opção booleana                                    :se[t] no{option}
trocar estado opção booleana                                :se[t] inv{option}
quebrar texto na coluna 72                                  :se tw=72
não quebrar                                                 :se tw=0
autoindent                                                  :se ai
expandir tab                                                :se et
especificar comentário (mail)                               :se comments=n:>,n:\|
```

```
executar GDB                    M-x gdb
descrever modo GDB              C-h m
passo uma linha                 M-s
próxima linha                   M-n
passo uma instrução (stepi)     M-i
finaliza stack frame atual      C-c C-f
continuar                       M-c
subir arg frames                M-u
descer arg frames               M-d
copiar número do ponto, inserir no final
                                C-x &
definir break point             C-x SPC
```

11.3.5 Configuração do Vim

Para usar todos os recursos do Vim e destaque de sintaxe, inclua as seguintes linhas em ~`/.vimrc` ou `/etc/vimrc`:

```
set nocompatible
set nopaste
set pastetoggle=<f11>
syn on
```

Modo Paste permite evitar que o autoindent interfira com operações de cortar e colar em um terminal console. Isso faz mais que um simples ”:set noai“.

Veja ‘Usando GnuPG com o Vim’ on page 247 para integração com GnuPG.

11.3.6 Ctags

`apt-get install exuberant-ctags` e execute `ctags` nos arquivos fontes. Digite `:tag` *`nome_da_função`* no Vim e irá para a linha onde *nome_da_função* começa. As tags funcionam para C, C++, Java, Python, e muitas outras linguagens.

O Emacs tem as mesmas capacidades ctags.

11.3.7 Converter uma tela com sintaxe destacada em fonte HTML

`so \$VIMRUNTIME/syntax/2html.vim` a partir do modo comando do Vim converterá texto destacado em texto HTML. Salve com `:w arquivo.html` e `:q`. Útil para código C, etc.

11.3.8 Dividir a tela com o `vim`

O `vim` pode editar vários arquivos em um ambiente de tela multi-dividida. Digite `:help usr_08.txt` para mais detalhes.

Para dividir a tela entre arquivos diferentes, digite no prompt de comando do vi:

```
:split outro-arquivo
:vsplit outro-arquivo
```

Ou na linha de comando:

```
$ vi -o arquivo1.txt arquivo2.txt   # Divide horizontalmente
$ vi -O arquivo1.txt arquivo2.txt   # Divide verticalmente
```

produzirá um vi multi-janela.

```
$ vimdiff file.txt~ file.txt         # verifica mudanças recentes de file.txt
$ vimdiff file.en.sgml file.fr.sgml  # verifica mudanças de tradução
$ gvimdiff file.txt~ file.txt        # no X
```

produzirá uma agradável visão das diferenças entre um arquivo original e uma cópia de segurança. Em SGML ele combina tags, então a comparação de traduções neste modo funciona muito bem.

Movimentos especiais do cursor com comandos Ctrl-W:

```
Ctrl-W +      aumentar o tamanho de uma janela
Ctrl-W -      diminuir o tamanho de uma janela
Ctrl-W h      ir para a janela da esquerda
Ctrl-W j      ir para a janela abaixo
Ctrl-W k      ir para a janela acima
Ctrl-W l      ir para a janela da direita
...
```

Use o seguinte para controlar a rolagem da tela:

```
:set scrollbind
:set noscrollbind
```

Capítulo 12

Sistemas de Controle de Versão

12.1 Concurrent Versions System (CVS) (Sistema de Versões Concorrentes)

Confira `/usr/share/doc/cvs/html-cvsclient`, `/usr/share/doc/cvs/html-info`, `/usr/share/doc/cvsbook` com `lynx` ou rode `info cvs` e `man cvs` para informações detalhadas.

12.1.1 Instalando um servidor CVS

A configuração a seguir permitirá que apenas membros do grupo ″src“ submetam ao repositório CVS e apenas membros do grupo ″staff“ administrem o cvs, com isto reduzindo a probabilidade de que alguém atire no próprio pé.

```
# cd /var/lib; umask 002 ; sudo mkdir cvs # [Woody] FSH
# apt-get install cvs cvs-doc cvsbook
# export CVSROOT=/var/lib/cvs
# cd $CVSROOT
# chown root:src .  # "staff" para maior restrição a projetos novos.
# chmod 3775 .             # Se usou "staff" acima, use 2775 aqui.
# cvs -d /var/lib/cvs init # é mais seguro especificar -d explicitamente aqui!
# cd CVSROOT
# chown -R root:staff .
# chmod 2775 .
# touch val-tags
# chmod 664 history val-tags
# chown root:src history val-tags
```

12.1.2 Exemplos de sessão CVS

O que se segue configurará um ambiente shell para acesso ao repositório CVS.

CVS Anônimo (apenas para baixar)

Acesso remoto somente-leitura:

```
$ export CVSROOT=:pserver:anonymous@cvs.sf.net:/cvsroot/qref
$ cvs login
$ cvs -z3 co qref
```

Uso de servidor CVS local

Acesso local a partir de uma shell na mesma máquina:

```
$ export CVSROOT=/var/lib/cvs
```

Uso de servidor CVS remoto pserver

Acesso remoto sem SSH (uso da implementação do protocolo RSH no `cvs`):

```
$ export CVSROOT=:pserver:account@cvs.foobar.com:/var/lib/cvs
$ cvs login
```

Este uso é vulnerável a uma tocaia (eavesdropping attack).

Uso do CVS remoto via `ssh`

Acesso remoto com SSH:

```
$ export CVSROOT=:ext:account@cvs.foobar.com:/var/lib/cvs
```

ou para o SourceForge:

```
$ export CVSROOT=:ext:account@cvs.sf.net:/cvsroot/qref
```

Você pode usar autenticação RSA ('Conectando com menos senhas – RSA' on page 165) também. Isto elimina a espera pela senha (prompt password).

Criar um arquivo CVS novo

Para,

ITEM	VALOR	SIGNIFICADO
source tree:	*~/project-x*	Todos os códigos-fonte
Project name:	*project-x*	Nome do projeto
Vendor Tag:	*Main-branch*	marcador para toda a ramificação
Release Tag:	*Release-initial*	marcador para um lançamento específico

Então,

```
$ cd ~/projeto-x                 # acesse o diretório fonte
 ... crie a árvore dos fontes ...
$ cvs import -m "Início projeto-x" projeto-x Main-branch Release-initial
$ cd ..; rm -R ~/projeto-x
```

Trabalhar com CVS

Trabalhar com *projeto-x* usando o repositório CVS local:

```
$ cd                            # acesse a área de trabalho
$ cvs co projeto-x              # pegue os fontes do CVS para local
$ cd projeto-x
 ... faça mudanças no conteúdo ...
$ cvs diff -u                   # similar ao diff -u repository/ local/
$ cvs up -C arquivo_modificado       # desfaz mudanças em um arquivo
$ cvs ci -m "Descreva modificação"   # salve os fontes locais para o CVS
$ vi arquivonovo_adicionado
$ cvs add arquivonovo_adicionado
$ cvs ci -m "Adicionado arquivonovo_adicionado"
$ cvs up                        # mescle a última versão a partir do CVS
 ... para criar todos os subdiretórios recém criados do CVS, use
 ... ao invés "cvs up -d -P"
 ... atente para linhas que iniciam com "C nomearquivo"
 ... código não alterado é movido para `.#nomearquivo.versão'
 ... procure "<<<<<<<" e ">>>>>>>" em nomearquivo
$ cvs tag Release-1             # adicione marcador de lançamento
 ... edite mais ...
$ cvs tag -d Release-1          # remova marcador de lançamento
$ cvs ci -m "mais comentários"
$ cvs tag Release-1             # readicione marcador de lançamento
$ cd                            # volte à área de trabalho
$ cvs co -r Release-inicial -d antigo projeto-x
```

```
 ... mova a versão original para diretório antigo
$ cd antigo
$ cvs tag -b Release-initial-bugfixes # crie marcador de ramificação (-b)
 ... agora você pode trabalhar na versão antiga (Tag=sticky)
$ cvs update -d -P                      # não cria diretórios vazios
 ... a árvore dos fontes agora tem o marcador de fixação (sticky tag) "Release-ini
 ... trabalhe nesta ramificação
$ cvs up -d -P # sincronize com arquivos modificados por outros nesta ramificação
$ cvs ci -m "check nesta ramificação"
$ cvs update -kk -A -d -P
 ... remova o marcador de fixação e esqueça do conteúdo
 ... atualize a partir do tronco principal sem expansão de palavra-chave
$ cvs update -kk -d -P -j Release-initial-bugfixes
 ... Mescle a ramificação Release-initial-bugfixes com o tronco
 ... principal sem expansão de palavra-chave. Corrija conflitos com o editor
$ cvs ci -m "mesclando Release-initial-bugfixes"
$ cd
$ tar -cvzf antigo-projeto-x.tar.gz antigo  # arquive, -j para bz2
$ cvs release -d antigo           # remova fonte local (opcional)
```

Opções legais que convém lembrar (use-a(s) como primeiro(s) argumento(s) para `cvs`):

```
-n      simula as ações, sem executá-las realmente
-t      exibe mensagens que mostram os passos da atividade do CVS
```

Exportar arquivos a partir do CVS

Para obter a última versão do CVS, use ”tomorrow“:

```
$ cvs ex -D tomorrow nome_módulo
```

Administrar o CVS

Adicione apelidos a um projeto (servidor local):

```
$ su - admin           # um membro do staff
$ export CVSROOT=/var/lib/cvs
$ cvs co CVSROOT/modules
$ cd CVSROOT
$ echo "px -a projeto-x" >>modules
$ cvs ci -m "Agora px é um apelido para projeto-x"
$ cvs release -d .
$ exit                 # ou Ctrl-D para voltar a partir de su
```

```
$ cvs co -d projeto px
 ... baixa projeto-x (alias:px) a partir do CVS
 ... para o diretório do projeto
$ cd projeto
 ... faça mudanças no conteúdo ...
```

12.1.3 Resolvendo problemas do CVS

Permissões de arquivo no repositório

CVS não sobrescreverá o atual arquivo do repositório, antes o substituirá por outro. Assim, *permissão de escrita no diretório do repositório* é algo crítico. Para cada novo repositório criado, se necessário rode os seguintes comandos para ter certeza desta condição.

```
# cd /var/lib/cvs
# chown -R root:src repository
# chmod -R ug+rwX   repository
# chmod    2775     repository  # se necessário, também o subdiretório
```

Bit de execução

O bit de execução de um arquivo é mantido quando baixado. Sempre que encontrar problemas com permissão de execução em arquivos baixados, mude as permissões do arquivo no repositório CVS com o seguinte comando.

```
# chmod ugo-x nomedoarquivo
```

12.1.4 Comandos CVS

Aqui estão os comandos CVS com suas abreviações

```
{add|ad|new} [-k kflag] [-m 'mensagem'] arquivos...
{admin|adm|rcs} [opções-rcs] arquivos...
{annotate|ann} [opções] [arquivos...]
{checkout|co|get} [opções] módulos...
{commit|ci|com}   [-lnR]  [-m  'mensagem_log'  |  -f  arquivo] \
        [-r revisão] [arquivos...]
{diff|di|dif} [-kl] [opções_rcsdiff] [[-r rev1 | -D data1] \
        [-r rev2 |  -D data2]] [arquivos...]
{export|ex|exp} [-flNn] -r rev|-D data [-d dir] [-k kflag] módulo...
{history|hi|his} [-report] [-flags] [-options args] [arquivos...]
{import|im|imp} [-options] repositório vendortag releasetag...
```

```
{login|logon|lgn}
{log|lo|rlog} [-l] opções-rlog [arquivos...]
{rdiff|patch|pa} [-flags] [-V vn] [-r t|-D d [-r t2|-D d2]] módulos...
{release|re|rel} [-d] diretórios...
{remove|rm|delete} [-lR] [arquivos...]
{rtag|rt|rfreeze} [-falnR]  [-b]  [-d]  [-r  marcador  |  -D  data] \
         marcador_simbólico módulos...
{status|st|stat} [-lR] [-v] [arquivos...]
{tag|ta|freeze} [-lR] [-F] [-b] [-d] [-r marcador | -D data]  [-f] \
         marcador_simbólico [arquivos...]
{update|up|upd} [-AdflPpR] [-d] [-r marcador|-D data] arquivos...
```

12.2 Subversion

Subversion é a próxima geração em sistemas de controle de versão. E pretende-se que substitua o CVS. Os desenvolvedores o consideram atualmente em estágio ″alpha“, embora provavelmente seja estável para muitos usos. Quando este texto foi escrito, Subversion estava disponível na Debian `unstable` apenas.

12.2.1 Instalando um servidor Subversion

O meta-pacote `subversion-server` depende dos pacotes `libapache2-dav-svn` e `subversion-tools`) para configurar um servidor.

Configurando um repositório

Atualmente, o pacote `subversion` não configura automaticamente um repositório. Isto deve ser feito manualmente. Uma localização possível para o repositório é em `/var/local /repos`.

Crie o diretório:

```
# mkdir -p /var/local/repos
```

Crie o banco de dados do repositório:

```
# svnadmin create /var/local/repos
```

Torne o repositório passível de escrita pelo servidor WWW:

```
# chown -R www-data:www-data /var/local/repos
```

Configurando Apache2

Para permitir acesso ao repositório via autenticação de usuário, adicione (ou descomente) o seguinte a `/etc/apache2/mods-available/dav_svn.conf`:

```
<Location /repos>
  DAV svn
  SVNPath /var/local/repos
  AuthType Basic
  AuthName "Subversion repository"
  AuthUserFile /etc/subversion/passwd
  <LimitExcept GET PROPFIND OPTIONS REPORT>
    Require valid-user
  </LimitExcept>
</Location>
```

Então, crie o arquivo de autenticação do usuário com o comando:

```
htpasswd2 -c /etc/subversion/passwd algum-nome-de-usuario
```

Reinicie Apache2, e seu novo repositório Subversion será acessível através da URL http://*hostname*/repos.

12.2.2 Movendo um repositório CVS para Subversion

12.2.3 Exemplos de uso do Subversion

As seções seguintes ensinam como usar diferentes comandos em Subversion.

Criar um novo repositório Subversion

Para criar um novo repositório Subversion, digite o seguinte:

```
$ cd ~/seu-projeto          # vá para seu diretório fonte
$ svn import http://localhost/repos seu-projeto \
  nome-do-projeto -m "importação inicial do projeto"
```

Isso cria um diretório chamado *nome-do-projeto* em seu repositório Subversion que contém os arquivos de seu projeto. Olhe em http://localhost/repos e veja se estão lá.

Trabalhando com Subversion

Trabalhando com *projeto-y* usando Subversion:

```
$ cd                              # vá para a área de trabalho
$ svn co http://localhost/repos/projeto-y  # Importe os fontes
$ cd projeto-y
 ... faça alguma coisa ...
$ svn diff                        # similar a diff -u repositório/ local/
$ svn revert arquivo_modificado   # desfaz mudanças em um arquivo
$ svn ci -m "Descreva alterações" # exporte suas alterações ao repositório
$ vi arquivonovo_adicionado
$ svn add arquivonovo_adicionado
$ svn add novo_dir                # adiciona recursivamente todos arquivos em novo_
$ svn add -N novo_dir2            # adiciona o diretório não recursivamente
$ svn ci -m "Adicionado arquivonovo_adicionado, novo_dir, novo_dir2
$ svn up                          # mescla com última versão do repositório
$ svn log                         # exibe todas as mudanças enviadas
$ svn copy http://localhost/repos/projeto-y \
      http://localhost/repos/projeto-y-ramo \
      -m "criando nova ramificação do projeto-y" # ramificando projeto-y
$ svn copy http://localhost/repos/projeto-y \
      http://localhost/repos/proj-y_versão1.0 \
      -m "projeto-y versão 1.0"   # adicionado marcador de lançamento
 ... note que a ramificação e marcação são os mesmos. A única diferença
 ... é que ramos (branches) são enviados (committed) enquanto
 ... marcadores (tags) não são.

 ... faça modificações no ramo ...

$ # mesclar cópia ramificada de volta na cópia principal
$ svn merge http://localhost/repos/projeto-y \
   http://localhost/repos/projeto-y-ramo
$ svn co -r 4 http://localhost/repos/projeto-y # importa revisão 4
```

Capítulo 13

Programação

Não use "test" como o nome de um arquivo de teste executável. O `test` é um comando interno de shell.

13.1 Onde começar

Referências:

- Documentos e exemplos sob o `/usr/share/doc/`*`pacote`*
- Unix / Informação sobre Programação (`http://arioch.unomaha.edu/~jclark/#info`)
- *Linux Programming Bible* (John Goerzen/IDG books)

Muitos documentos informativos longos podem ser obtidos como brochuras do GNU (`http://www.gnu.org/`).

As próximas quatro seções contêm scripts de exemplo em diferentes linguagens para criar um arquivo texto de informação de uma conta para ser adicionado no `/etc/passwd` usando um processador de arquivo de lote como o programa `newusers`. Cada script requer como entrada um arquivo com linhas na forma `prenome sobrenome senha`. (Os diretórios home do usuário real não serão criados através desses scripts.)

13.2 Shell

Ler scripts shell é a **melhor** maneira de entender como um sistema similar ao Unix funciona. Aqui, dou alguns apontamentos e lembretes de programação shell. Veja Erros de Shell (`http://www.greenend.org.uk/rjk/2001/04/shell.html`) para aprender a partir de erros.

13.2.1 Bash – o shell interativo padrão do GNU

Referências sobre o Bash:

- `bash(1)`
- `info bash`
- o BASH Programming - Introduction HOWTO (`http://www.tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html`) do LDP como informção inicial.
- `mc /usr/share/doc/bash/examples/ /usr/share/doc/bash/` (Instale o pacote `bash-doc` para ver os arquivos de exemplo.)
- *Learning the bash Shell*, segunda edição (O'Reilly)

Programa de exemplo curto (cria entradas de conta para o `newusers` a partir da entrada padrão):

```
#!/bin/bash
# (C) Osamu Aoki Sun Aug 26 16:53:55 UTC 2001 Public Domain
pid=1000;
while read n1 n2 n3 ; do
if [ ${n1:0:1} != "#" ]; then
let pid=$pid+1
echo ${n1}_${n2}:password:${pid}:${pid}:,,,/home/${n1}_${n2}:/bin/bash
fi
done
```

13.2.2 Shells POSIX

Vários pacotes fornecem um shell POSIX no Debian:

- `dash` (Sarge)
  - Prioridade: opcional
  - Tamanho instalado: 176
  - O menor de todos e de longe o mais rápido – o melhor para primeira inicialização
- `ash` (Woody)
  - Prioridade: opcional
  - Tamanho instalado: 180
  - Menor e muito mais rápido – bom para primeira inicialização
- `bash`
  - Essencial: sim
  - Prioridade: requerido
  - Tamanho instalado: 580
  - Maior e com mais recursos – muitas extensões implementadas
- `pdksh`
  - Prioridade: opcional
  - Tamanho instalado: 408
  - Similar do ksh AT&T completo

Se você estiver escrevendo um script shell para portabilidade, é melhor escrevê-lo como um script shell POSIX. Use o `/bin/sh` ligado ao `ash` (ou `dash`) para testar sua compatibilidade POSIX. Evite escrever scripts com **bashismo** ou o **zshismo** que parece assemelhar-se à sintaxe de `csh`. Por exemplo, evite:

- `if [ foo == bar ] ; then ...`
- `diff -u arquivo.c{.orig,}`
- `mkdir /foo{bar,baz}`

A descrição para o shell nesse documento aplica-se somente aos shells tipo POSIX e assim não se aplica aos shells tipo `csh` incluindo `tcsh`.

13.2.3 Parâmetros shell

Há vários **parâmetros especiais** a lembrar:

```
$0      = nome do shell ou do script shell
$1      = primeiro(1) argumento do shell
 ...
$9      = nono(9) argumento do shell
$#      = número de parâmetros posicionais
"$*"    = "$1 $2 $3 $4 ... $n"
"$@"    = "$1" "$2" "$3" "$4" ... "$n"
$?      = estado de saída do comando mais recente
$$      = PID desse script shell
$!      = PID do trabalho em background iniciado mais recentemente
```

Expansões de parâmetros básicas a lembrar:

Forma	Se *var* está def.(*)	Se *var* não está def.(*)
`${var:-string}`	`$var`	*string*
`${var:+string}`	*string*	`null`
`${var:=string}`	`$var`	*string* (e executa *var*=*string*)
`${var:?string}`	`$var`	(echo *string* e então sai)

Aqui, os dois pontos ':' em todos esses operadores são atualmente opcionais.

- Com ':' = operador de teste para "existe" e "não nulo".
- Sem ':' = operador de teste somente para "existe".

Substituições de parâmetros básicas a lembrar:

Forma	Resultado
`${var%sufixo}`	Remove menor padrão *sufixo*
`${var%%sufixo}`	Remove maior padrão *sufixo*
`${var#prefixo}`	Remove menor padrão *prefixo*
`${var##prefixo}`	Remove maior padrão *prefixo*

13.2.4 Redirecionamento shell

Redirecionamentos básicos a lembrar (aqui o [*n*] é um número opcional para especificar o descritor de arquivo):

```
[n]> arquivo   Redireciona stdout (ou n) para o arquivo.
[n]>> arquivo  Adiciona a stdout (ou n) ao arquivo.
[n]< arquivo   Redireciona stdin (ou n) a partir do arquivo.
[n1]>&n2       Redireciona stdout (ou n1) para n2.
2> arquivo >&2  Redireciona stdout e stderr para o arquivo.
| comando      Faz um pipe da stdout para o comando.
2>&1 | comando  Faz um pipe da stderr e stdout para o comando.
```

Aqui,
- stdin: entrada padrão (descritor de arquivo = 0)
- stdout: saída padrão (descritor de arquivo = 1)
- stderr: erro padrão (descritor de arquivo = 2)

O shell possibilita que você abra arquivos usando o comando interno `exec` com um descritor de arquivo arbitrário.

```
$ echo Olá >foo
$ exec 3<foo 4>bar  # abre arquivosopen files
$ cat <&3 >&4       # redireciona stdin p/ 3, stdout p/ 4
$ exec 3<&- 4>&-    # fecha arquivos
$ cat bar
Olá
```

Aqui, *n*`<&-` e *n*`>&-` significam fechar o descritor de arquivo *n*.

13.2.5 Condicionais de shell

Cada comando retorna um **estado de saída** que pode ser usado para expressões condicionais:
- Successo: 0 (Verdadeiro)
- Erro: 1–255 (Falso)

Note que o uso do valor 0 aqui para significar "verdadeiro" difere da convenção normal de outras áreas da computação. Além disso, '[' é o equivalente do comando `test`, que avalia seus argumentos até o ']' como uma expressão condicional.

As **sentenças condicionais** básicas a lembrar são:

```
comando && se_sucesso_executa_esse_comando_também || true
command || se_insucesso_ao_invés_executa_esse_comando

if [ expressão_condicional ]; then
 se_sucesso_executa_esse_comando
else
 se_insucesso_executa_esse_comando
fi
```

Aqui o `|| true` foi necessário para garantir que esse script shell não finalize nessa linha acidentalmente quando o shell for invocado usando com a opção `-e`.

Os operadores de comparação de **arquivos** nas expressões condicionais são:

```
-e arquivo             Verdadeiro se o arquivo existe.
-d arquivo             Verdadeiro se o arquivo existe e é um diretório.
-f arquivo             Verdadeiro se o arquivo existe e é um arquivo regular.
-w arquivo             Verdadeiro se o arquivo existe e é gravável.
-x arquivo             Verdadeiro se o arquivo existe e é executável.
arquivo1 -nt arquivo2 Verdadeiro se o arquivo1 é mais novo que o arquivo2. (modifi
arquivo1 -ot arquivo2 Verdadeiro se o arquivo1 é mais antigo que o arquivo2. (modi
arquivo1 -ef arquivo2 Verdadeiro se eles são o mesmo dispositivo e número de inode
```

Os operadores de comparação de **cadeias de caracteres** em expressões condicionais são:

```
     -z str    Verdadeiro se o comprimento de str é zero.
     -n str    Verdadeiro se o comprimento de str não é zero.
str1 == str2   Verdadeiro se as cadeias de caracteres são iguais.
str1 = str2    Verdadeiro se as cadeias de caracteres são iguais.
    ( "=" deve ser usado no lugar de "==" para conformidade POSIX estrita )
str1 != str2   Verdadeiro se as cadeias de caracteres não são iguais.
str1 <  str2   Verdadeiro se str1 é ordenada antes de str2 (depende do locale).
str1 >  str2   Verdadeiro se str1 é ordenada depois de str2 (depende do locale).
```

Os operadores de comparação **aritmética** inteira nas expressões condicionais são `-eq`, `-ne`, `-lt`, `-le`, `-gt`, e `-ge`.

13.2.6 Processamento de linha de comando

O shell processa um script como a seguir:

- divide em **tokens** através dos metacaracteres: ESPAÇO, TAB, NOVALINHA, ;, (,), <, >, `|`, &
- verifica a **palavra-chave** se não estiver entre ”. . . “ ou ’. . . ’ (loop)
- expande **alias** se não estiver entre ”. . . “ ou ’. . . ’ (loop)
- expande **chaves**, `a{1,2}` -> `a1 a2`, se não estiver entre ”. . . “ ou ’. . . ’
- expande **til**, *~usuario* -> diretório home do *usuario*, se não estiver entre ”. . . “ or ’. . . ’
- expande **parâmetro**, $*PARAMETER*, se não estiver entre ’. . . ’
- expande **substituição de comando**, $(*comando*), se não estiver entre ’. . . ’
- divide em **palavras** com $IFS se não estiver entre ”. . . “ ou ’. . . ’
- expande *?[] em **pathname** se não estiver entre ”. . . “ ou ’. . . ’
- procura o **comando**
  - função
  - comando interno
  - arquivo em $PATH
- loop

As aspas simples dentro de aspas duplas não têm efeito.

Executar `set -x` no shell ou invocá-lo com a opção `-x` faz com que o shell imprima todos os comandos executados. Isso é um tanto útil para depuração.

13.3 Awk

Referências sobre o Awk:

- *Effective awk Programming*, terceira edição (O'Reilly)
- *Sed & awk*, segunda edição (O'Reilly)
- `mawk(1)` e `gawk(1)`
- `info gawk`

Programa de exemplo curto (cria entrada para o comando `newusers`):

```
#!/usr/bin/awk -f
# Script para criar um arquivo adequado para usar no comando 'newusers',
# a partir de um arquivo consistindo de identificações de usuários e senhas na form
# Prenome Sobrenome senha
# Copyright (c) KMSelf Sat Aug 25 20:47:38 PDT 2001
# Distribuído sob a GNU GPL v 2, ou sob sua opção, qualquer versão posterior.
# Este programa é distribuído SEM QUALQUER GARANTIA.

BEGIN {
    # Atribui UID, GID que está iniciando
    if ( ARGC > 2 ) {
         startuid = ARGV[1]
         delete ARGV[1]
    }
    else {
        printf( "Uso:  newusers startUID arquivo\n" \
           "  onde:\n"\
           "    startUID é o ID do usuário a adicionar, e\n" \
           "    arquivo é um arquivo de entrada na forma:\n" \
           "      prenome sobrenome senha\n" \
        )
        exit
    }

    infile = ARGV[1]
    printf( "Iniciando UID: %s\n\n", startuid )
}

/^#/ { next }

{
    ++record
    first = $1
    last = $2
    passwd = $3
    user= substr( tolower( first ), 1, 1 ) tolower( last )
```

```
        uid = startuid + record - 1
        gid = uid
        printf( "%s:%s:%d:%d:%s %s,,/home/%s:/bin/bash\n",  \
             user, passwd, uid, gid, first, last, user \
             )
}
```

Há dois pacotes que fornecem um `awk` POSIX no Debian:

- `mawk`
  - Prioridade: requerido
  - Tamanho instalado: 228
  - Menor e muito mais rápido – bom para instalação padrão
  - Existem limites na compilação
    - NF = 32767
    - sprintf buffer = 1020
- `gawk`
  - Prioridade: opcional
  - Tamanho instalado: 1708
  - Maior e com muitos recursos – muitas extensões implementadas
    - versão de UNIX System V Release 4
    - awk da Bell Labs
    - específico GNU

13.4 Perl

Este é **o** interpretador em um sistema similar ao Unix.

Referências sobre o Perl:

- `perl(1)`
- *Programming Perl,* terceira edição (O'Reilly)

Programa de exemplo curto (cria entrada para o comando `newusers`):

```
#!/usr/bin/perl
# (C) Osamu Aoki Sun Aug 26 16:53:55 UTC 2001 Public Domain
$pid=1000;
while (<STDIN>) {
        if (/^#/) { next;}
        chop;
        $pid++;
        ($n1, $n2, $n3) = split / /;
        print $n1,"_",$n2,":", $n3, ":",$pid,
                  ":",$pid,",,,/home/",$n1,"_",$n2,":/bin/bash\n"
}
```

Para instalar o módulo Perl *nome_módulo*:

```
# perl -MCPAN -e 'install nome_módulo'
```

13.5 Python

Esse é um interpretador orientado a objeto muito bom.

Referências sobre o Python:

- `python(1)`
- *Learning Python* (O'Reilly).

Programa de exemplo curto (cria entrada para o comando `newusers`):

```
#! /usr/bin/env python
import sys, string

# (C) Osamu Aoki Sun Aug 26 16:53:55 UTC 2001 Public Domain
# Portado do script awk de KMSelf Sat Aug 25 20:47:38 PDT 2001
# Este programa é distribuído SEM QUALQUER GARANTIA.

def usages():
    print \
"Uso:  ", sys.argv[0], " start_UID [nome_de_arquivo]\n" \
"\tstartUID é o ID de usuário iniciando a adicionar.\n" \
"\tnome_de_arquivo é o nome do arquivo de entrada. Se não especificado, entrada pa
"Formato do arquivo de entrada:\n"\
"\tprenome sobrenome senha\n"
                 return 1

def parsefile(startuid):
    #
    # filtro principal
    #
    uid = startuid
    while 1:
        line = infile.readline()
        if not line:
            break
        if line[0] == '#':
            continue
        (first, last, passwd) = string.split(string.lower(line))
        # acima falha com número errado de parâmetros :-)
        user = first[0] + last
        gid = uid
```

```
        lineout = "%s:%s:%d:%d:%s %s,,/home/%s:/bin/bash\n" %  \
            (user, passwd, uid, gid, first, last, user)
        sys.stdout.write(lineout)
        +uid

if __name__ == '__main__':
    if len(sys.argv) == 1:
        usages()
    else:
        uid = int(sys.argv[1])
        #print "# UID start from: %d\n" % uid
        if len(sys.argv) > 1:
            infilename   = string.join(sys.argv[2:])
            infile = open(infilename, 'r')
            #print "# Read file from: %s\n\n" % infilename
        else:
            infile = sys.stdin
        parsefile(uid)
```

13.6 Make

Referências sobre o Make:

- `info make`
- `make(1)`
- *Managing Projects with make*, segunda edição (O'Reilly)

Variáveis automáticas simples:

Sintaxe de regra:

```
alvo: [ pré-requisitos ... ]
 [TAB]  comando1
 [TAB]  -comando2 # ignora erros
 [TAB]  @comando3 # suprime o eco
```

Aqui, `[TAB]` é um código TAB. Cada linha é interpretada pelo shell depois de fazer a substituição de variável. Use \ no fim de uma linha para continuar o script. Use `$$` para entrar `$` para valores de ambiente para um script shell.

Regras implícitas para o *alvo* e *pré-requisitos* podem ser escritas, por exemplo, como:

```
%: %.c header.h
```

ou

```
%.o: %.c header.h
```

Aqui, o *alvo* contém o caracter `%` (exatamente um deles). O `%` pode corresponder a qualquer sub-cadeia de caracteres não vazia nos nomes de arquivo do alvo atual. Os *pré-requisitos* da mesma forma usam `%` para mostrar como seus nomes se relacionam com o nome do alvo atual.

Regras de sufixo são a forma **obsoleta** de definir regras implícitas para o `make`. Elas ainda são suportadas no GNU `make` por compatibilidade, mas use regras de padrão equivalente, sempre que possível:

```
regra de sufixo antiga --> nova regra de padrão
.c:                    --> %  : %.c
.c.o:                  --> %.o: %.c
```

Variáveis automáticas para a regra:

```
foo.o: new1.c new2.c old1.c new3.c
$@ == foo.o                         (alvo)
$< == new1.c                        (o primeiro)
$? == new1.c new2.c new3.c          (os mais novos)
$^ == new1.c new2.c old1.c new3.c   (todos)
$* == `%' padrão correspondente no padrão alvo.
```

Referências variáveis:

```
foo1 := bar    # Expansão única
foo2  = bar    # Expansão recursiva
foo3 += bar    # Adiciona
SRCS := $(wildcard *.c)
OBJS := $(foo:c=o)
OBJS := $(foo:%.c=%.o)
OBJS := $(patsubst %.c,%.o,$(foo))
DIRS  = $(dir diretorio/nomearquivo.ext) # Extrai "diretorio"
$(notdir NAMES...), $(basename NAMES...), $(suffix NAMES...) ...
```

Execute `make -p -f/dev/null` para ver as regras internas automáticas.

13.7 C

Preparação:

```
# apt-get install glibc-doc manpages-dev libc6-dev gcc
```

Referências para o C:

- `info libc` (referência de função da biblioteca C)
- `gcc(1)`
- `nome_de_cada_função_da_biblioteca_C(3)`
- Kernighan & Ritchie, *The C Programming Language*, segunda edição (Prentice Hall).

13.7.1 Programa C simples (`gcc`)

Um exemplo simples para compilar `example.c` com uma biblioteca `libm` em um executável `run_example`:

```
$ cat > example.c << EOF
#include <stdio.h>
#include <math.h>
#include <string.h>

int main(int argc, char **argv, char **envp){
        double x;
        char y[11];
        x=sqrt(argc+7.5);
        strncpy(y, argv[0], 10); /* evita buffer overflow */
        y[10] = '\0'; /* preenche para ter certeza que string termina com '\0' */
        printf("%5i, %5.3f, %10s, %10s\n", argc, x, y, argv[1]);
        return 0;
}
EOF
$ gcc -Wall -g -o run_example example.c -lm
$ ./run_example
        1, 2.915, ./run_exam,     (null)
$ ./run_example 1234567890qwerty
        2, 3.082, ./run_exam, 1234567890qwerty
```

Aqui, a opção `-l`**`m`** é necessária para ligar a biblioteca `lib`**`m`** para o `sqrt()`. A biblioteca real está no `/lib/` com o nome de arquivo `libm.so.6`, que é um link simbólico para o `libm-2.1.3.so`.

Veja o último parâmetro no texto de saída. Há mais de 10 caracteres apesar de `%10s` ser especificado.

O uso de funções que realizam operações de apontadores sem verificações de limite, como o `sprintf` e o `strcpy`, está ultrapassado pois não elas evitam as vulnerabilidades causadas por buffer overflow que leva aos efeitos acima. Ao invés disso, use `snprintf` e `strncpy`.

13.7.2 Depurando

Depurando com o `gdb`

Preparação:

```
# apt-get install gdb
```

Referências sobre o `gdb`:

- `info gdb` (tutorial)
- `gdb(1)`
- `http://www.unknownroad.com/rtfm/gdbtut/gdbtoc.html`

Use o `gdb` para depurar um programa compilado com a opção `-g`. Muitos comandos podem ser abreviados. A expansão de tab funciona como no shell.

```
$ gdb programa
(gdb) b 1                # define ponto de parada na linha 1
(gdb) run arg1 arg2 arg3 # executa o programa
(gdb) next               # próxima linha
...
(gdb) step               # avança um passo
...
(gdb) p parm             # imprime parm
...
(gdb) p parm=12          # define o valor para 12
```

Para depurar a partir do Emacs, consulte o 'Resumo de comandos do Editor (Emacs, Vim)' on page 213.

Verificar dependências de bibliotecas

Use o `ldd` para determinar as dependências de bibliotecas de um programa:

```
$ ldd /bin/ls
        librt.so.1 => /lib/librt.so.1 (0x4001e000)
        libc.so.6 => /lib/libc.so.6 (0x40030000)
        libpthread.so.0 => /lib/libpthread.so.0 (0x40153000)
        /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Para o `ls` funcionar em um ambiente usando `chroot`, as bibliotecas acima precisam estar disponíveis no ambiente `chroot`.

Os seguintes comandos também serão úteis:

- `strace`: rastrea chamadas de sistema e sinais
- `ltrace`: rastrea chamadas de biblioteca

Depurando com ferramentas de detecção de fuga de memória

Há várias ferramentas de detecção de fuga de memória disponíveis no Debian.

- `njamd`
- `valgrind`
- `dmalloc`
- `electric-fence`
- `memprof`
- `memwatch` (não empacotado, obtenha-o em memwatch (`http://directory.fsf.org/devel/debug/memwatch.html`).)
- `mpatrol`
- `leaktracer`
- `libgc6`
- `Insure++` da Parasoft (`http://www.parasoft.com`). (não livre, comercial)

Verifique também as Ferramentas de Depuração para Alocação Dinâmica de Armazenamento e Gerenciamento de Memória (`http://www.cs.colorado.edu/homes/zorn/public_html/MallocDebug.html`).

13.7.3 Flex – um Lex melhor

O `flex` é um gerador de analisador léxico rápido.

Referências sobre o `flex`:

- `info flex` (tutorial)
- `flex(1)`

Você precisa fornecer suas próprias funções `main()` e `yywrap()`, ou seu `programa.l` deve parecer assim para compilar sem uma biblioteca (`yywrap` é um macro; `%option main` ativa `%option noyywrap` implicitamente):

```
%option main
%%
.|\n     ECHO ;
%%
```

Alternativamente, você pode compilar com a opção `-lfl` do ligador no fim da sua linha de comando `cc` (como o AT&T-Lex com `-ll`). Não é necessário `%option` nesse caso.

13.7.4 Bison – um Yacc melhor

Há vários pacotes que fornecem um gerador de analisador LALR compatível com Yacc no Debian:

- `bison`: Gerador de analisador LALR GNU
- `byacc`: O gerador de analisador LALR Berkeley

- `btyacc`: Gerador de analisador reverso baseado no `byacc`

Referências sobre o `bison`:

- `info bison` (tutorial)
- `bison(1)`

Você precisa fornecer suas próprias funções `main()` e `yyerror()`. A função `main()` chama a `yyparse()` que chama a `yylex()`, geralmente criada com o FleX.

```
%%

%%
```

13.7.5 Autoconf

O `autoconf` é uma ferramenta para produzir scripts shell que configuram automaticamente pacotes de código fonte de software para adaptá-lo às muitas formas de sistemas similares ao UNIX usando o sistema de criação GNU inteiro.

O `autoconf` produz o script de configuração `configure`. O `configure` cria automaticamente um `Makefile` personalizado usando o modelo `Makefile.in`.

Compilar e instalar um programa

O Debian não mexe em arquivos no `/usr/local` (veja 'Suportando diversidades' on page 23). Então se você compilar um programa a partir da fonte, instale-o no `/usr/local`, assim ele não interferirá com o Debian.

```
$ cd src
$ ./configure --prefix=/usr/local
$ make
$ make install # isso coloca os arquivos no sistema
```

Desinstalar um programa

Se você ainda tiver a fonte e o programa usar o `autoconf/automake` e se você puder lembrar-se de como o configurou:

```
$ ./configure todas-as-opções-que-você-deu
# make uninstall
```

Por outro lado, se você tem certeza absoluta de que o processo de instalação põe arquivos somente sob o `/usr/local` e não há nada importante lá, você pode apagar todo seu conteúdo com:

```
# find /usr/local -type f -print0 | xargs -0 rm -f
```

Se você não tiver certeza de onde os arquivos foram instalados, deve pensar em usar o `checkinstall`, que fornece uma maneira limpa para a desinstalação.

13.8 Preparação de documentos

13.8.1 Processamento de texto `roff`

Tradicionalmente, o `roff` é o principal sistema de processamento de texto do Unix.

Veja `roff(7)`, `groff(7)`, `groff(1)`, `grotty(1)`, `troff(1)`, `groff_mdoc(7)`, `groff_man(7)`, `groff_ms(7)`, `groff_me(7)`, `groff_mm(7)`, e `info groff`.

Há um bom tutorial sobre macros `-me`. Se você tiver o `groff` (1.18 ou mais novo), encontre o arquivo `/usr/share/doc/groff/meintro.me.gz` e faça o seguinte:

```
$ zcat /usr/share/doc/groff/meintro.me.gz | \
     groff -Tascii -me - | less -R
```

O seguinte fará um arquivo texto completamente plano:

```
$ zcat /usr/share/doc/groff/meintro.me.gz | \
    GROFF_NO_SGR=1 groff -Tascii -me - | col -b -x > meintro.txt
```

Para imprimir, use a saída PostScript.

```
$ groff -Tps meintro.txt | lpr
$ groff -Tps meintro.txt | mpage -2 | lpr
```

13.8.2 SGML

Preparação:

```
# apt-get install debiandoc-sgml debiandoc-sgml-doc
```

Referências sobre o `debiandoc-sgml`:

- `/usr/share/doc/debiandoc-sgml-doc`
- `debiandoc-sgml(1)`
- *DocBook: The Definitive Guide* (*`/usr/share/doc/docbook-defguide/html/docbook.html`*), por Walsh e Muellner (O'Reilly) (pacote `docbook-defguide`

O SGML possibilita o gerenciamento de vários formatos de um documento. Um sistema SGML fácil é o Debiandoc, que é usado aqui. Ele requer pequenas adaptações dos arquivos texto originais para os seguintes caracteres:

- ”<“ –> `<`
- ”>“ –> `>`
- ” “ –> ` ` (espaço inquebrável)
- ”&“ –> `&`
- ”%“ –> `%`
- ”©“ –> `©`
- ”–“ –> `–`
- ”—“ –> `—`

Para marcar uma seção como um comentário não imprimível, entre:

```
<!-- Indique o problema aqui ... -->
```

Para marcar uma seção com um comentário alterável, entre:

```
<![ %FIXME; [ Indique o problema aqui ... ]]>
```

Em SGML, a *primeira definição* de uma entidade vence. Por exemplo:

```
<!entity % qref "INCLUDE">
<![ %qref; [ <!entity param "Dado 1"> ]]>
<!entity param "Dado 2">
&param;
```

Isso finaliza como ”Dado 1“. Se a primeira linha tiver ”IGNORE“ ao invés de ”INCLUDE“, isso finaliza como ”Dado 2“ (a segunda linha é uma sentença condicional). Além disso, a repetição de frases pode ser definida antes separadamente do contexto.

```
<!entity quem-e-esse "meu">
Olá &quem-e-esse; amigo.
Esse é  &quem-e-esse; livro.
```

Isso resulta no seguinte:

```
Olá meu amigo.
Esse é meu livro.
```

Veja o exemplo curto de SGML `sample.sgml` nos exemplos (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

Quando os documentos SGML ficam maiores, algumas vezes o TeX que é usado como o processador de texto em plano de fundo (backend) pode causar erros. Veja ‘TeX/LaTeX’ on the next page.

13.8.3 TeX/LaTeX

Preparação:

```
# tasksel # selecione Miscellaneous  --> TeX/LaTeX environment
```

Referências para LaTeX:

- The teTeX HOWTO: The Linux-teTeX Local Guide (`http://www.tldp.org/HOWTO/TeTeX-HOWTO.html`)
- `tex(1)`
- `latex(1)`
- *The TeXbook*, de Donald E. Knuth, (Addison-Wesley) [1]
- *LaTeX - A Document Preparation System*, por Leslie Lamport, (Addison-Wesley)
- *The LaTeX Companion*, de Goossens, Mittelbach, Samarin, (Addison-Wesley)

Esse é o ambiente de editoração mais poderoso. Muitos processadores SGML o usam como seu processador de texto. O Lyx fornecido nos pacotes `lyx`, `lyx-xforms`, ou `lyx-qt` oferece um agradável ambiente de edição WYSIWYG para LaTeX enquanto muitos usam o Emacs e o Vim como alternativa para o editor de fonte.

Há muitos recursos online disponíveis.

- teTeX - A Documentation Guide (`/usr/share/doc/texmf/newhelpindex.html`) (pacote `tetex-doc`)
- The Not So Short Introduction to LaTeX2e (`ftp://ftp.dante.de/tex-archive/info/lshort/portuguese-BR/`)
- A Quick Introduction to LaTeX (`http://www.msu.edu/user/pfaffben/writings/`)
- A Simple Guide to Latex/Lyx (`http://www.stat.rice.edu/~helpdesk/howto/lyxguide.html`)
- Word Processing Using LaTeX (`http://www-h.eng.cam.ac.uk/help/tpl/textprocessing/latex_basic/latex_basic.html`)
- Local User Guide to teTeX/LaTeX (`http://supportweb.cs.bham.ac.uk/documentation/LaTeX/lguide/local-guide/local-guide.html`)

Quando os documentos ficam maiores, algumas vezes o TeX pode causar erros. Você precisa aumentar o tamanho do pool no `/etc/texmf/texmf.cnf` (ou mais apropriadamente, edite o `/etc/texmf/texmf.d/95NonPath` e execute `update-texmf`) para resolver isso.

13.8.4 Programação Letrada (Literate Programming)

Ao invés de escrever código contendo documentação, o programador letrado escreve documentação contendo código. Essa abordagem garante uma boa documentação para um pro-

---

[1] A fonte TeX desse livro está disponível em `ftp://ftp.dante.de/pub/tex/systems/knuth/tex/texbook.tex`. `ftp://ftp.dante.de/pub/tex/systems/knuth/lib/manmac.tex` contém muitas das macros necessárias. Você pode processar esse documento com o `tex` depois de comentar as linhas 7 até 10 e adicionar `\input manmac \proofmodefalse`. É fortemente recomendado comprar esse livro (e todos os outros livros de Donald E. Knuth) ao invés de usar a versão online mas a fonte é um grande exemplo de entrada TeX!

grama.

Para mais detalhes sobre programação letrada, veja Programação Letrada (`http://www.literateprogramming.com/`).

Noweb

Preparação:

```
# apt-get install nowebm
```

Referências para o Noweb:

- Noweb — Uma Ferramenta Simples, Extensível para Programação Letrada (`http://www.eecs.harvard.edu/~nr/noweb/`)
- `noweb(1)`

Essa é uma ferramenta de programação letrada semelhante a WEB que é mais simples mas fornece extensibilidade e independência de linguagem. [2] Quando o `noweb` é invocado, ele escreve o código fonte do programa nos arquivos de saída mencionados no arquivo noweb, e escreve um arquivo TeX para documentação editorada.

O pacote Debian `ifupdown` é um bom exemplo.

```
$ apt-get source ifupdown
$ cd ifupdown*
$ make ifupdown.pdf ifupdown.ps
```

Doxygen

Preparação:

```
# apt-get install doxygen doxygen-doc doxygen-gui
```

Referências para o Doxygen (criadas por `doxygen`!):

- Homepage (`http://www.doxygen.org/`)
- `/usr/share/doc/doxygen-doc/html/index.html`

Ele pode gerar documentação HTML, RTF, páginas de manual Unix, PostScript, e PDF (usando LaTeX) para programas em C++, C, Java, IDL e até certo ponto PHP e C#. O Doxygen é compatível com JavaDoc (1.1), Qt-Doc, KDOC e foi projetado especificamente para ser usado para projetos que fazem uso do toolkit Qt (`http://www.trolltech.no/qt/`) da Troll Tech. Ele cria gráficos de dependência de include, diagramas de colaboração, e gráficos de hierarquia de classe gráfica mesmo para programas não documentados. A saída é similar à documentação da Qt.

---

[2] Esse WEB não tem **nada** a ver com World Wide Web. WEB (para PASCAL) e CWEB (para C/C++) são ferramentas tradicionais para programação letrada.

13.9 Empacotamento

Preparação:

```
# apt-get install debian-policy developers-reference \
        maint-guide dh-make debhelper
# apt-get install packaging-manual # se for Potato
```

Referências sobre empacotamento:

- 'O sistema de gerenciamento de pacotes Debian' on page 11 (básico)
- Guia dos Novos Desenvolvedores Debian (tutorial)
- `dh-make(1)`
- Referência dos Desenvolvedores Debian (prática melhor)
- Manual de Políticas Debian (oficial)
- Manual de Empacotamento (Potato)

13.9.1 Empacotando um binário simples

Método curto e grosso para empacotar um binário simples de Joey Hess.

```
# mkdir -p mypkg/usr/bin mypkg/DEBIAN
# cp binary mypkg/usr/bin
# cat > mypkg/DEBIAN/control
Package: mypackage
Version: 1
Architecture: i386
Maintainer: Joey Hess <joeyh@debian.org>
Description: my little package
 Don't expect much.
^D
# dpkg-deb -b mypkg
```

13.9.2 Empacotando com ferramentas

Use o `dh_make` do pacote `dh-make` para criar um pacote inicial. Então, continue de acordo com as instruções de `dh-make(1)`. Ele usa o `debhelper` no `debian/rules`.

Uma alternativa mais antiga é usar o `deb-make` do pacote `debmake`. Ele não usa os scripts `debhelper` e depende apenas do shell.

Se quiser exemplos de pacotes de várias fontes, veja o "mc" (`dpkg-source -x mc_4.5.54.dsc`), que usa o "sys-build.mk" de Adam Heath (`<doogie@debian.org>`), e a "glibc" (`dpkg-source -x glibc_2.2.4-1.dsc`), que usa outro sistema do falecido Joel Klecker (`<espy@debian.org>`).

Capítulo 14

GnuPG

Referências:

- `gpg(1)`.
- `/usr/share/doc/gnupg/README.gz`
- *GNU privacy handbook* em `/usr/share/doc/gnupg-doc/GNU_Privacy_Handbook/` (instale `gnupg-doc`)

14.1 Instalando GnuPG

```
# gpg --gen-key                     # gera uma nova chave
# gpg --gen-revoke my_user_ID       # gera uma chave de revogação para my_user_ID
# host -l pgp.net | grep www|less # lista servidores de chaves pgp
```

Por ora, bons servidores de chave são:

```
keyserver wwwkeys.eu.pgp.net
keyserver wwwkeys.pgp.net
```

Deve-se ter cuidado aqui: **não crie mais do que duas subchaves**. Se você fizer isto, os servidores de chaves em pgp.net **corromperão** a sua chave. Use o novo `gnupg` (>1.2.1-2) para lidar com essas subchaves corrompidas. Consulte `http://fortytwo.ch/gpg/subkeys`.

E também apenas um servidor de chaves pode ser especificado em `$HOME/.gnupg/options`.

Infelizmente, os servidores abaixo não funcionam mais:

```
keyserver search.keyserver.net
keyserver pgp.ai.mit.edu
```

14.2 Usando GnuPG

Lidando com arquivos:

```
$ gpg  [opções]  comando  [args]
$ gpg {--armor|-a} {--sign|-s} arquivo # assina arquivo em um arquivo.asc texto
$ gpg --clearsign arquivo              # assina textualmente a mensagem
$ gpg --clearsign --not-dash-escaped arq_corr  # assina textualmente o arquivo de
$ gpg --verify arquivo                 # verifica o arquivo assinado textualmente
$ gpg -o arquivo.sig {-b|--detach-sig} arquivo # cria uma assinatura destacada
$ gpg --verify arquivo.sig arquivo     # verifica arquivo com arquivo.sig
$ gpg -o arq_cript {--recipient|-r} nome {--encrypt|-e} arquivo
        # criptografia com a chave pública para nome
$ gpg -o arq_cript {--symmetric|-c} arquivo # criptografia simétrica
$ gpg -o arquivo --decrypt arq_cript   # descriptografia
```

14.3 Gerenciando GnuPG

Gerenciamento das chaves:

```
$ gpg --edit-key ID_usr                # interativo; para ajuda, "help"
$ gpg -o arquivo --exports             # exporta todas as chaves para arquivo
$ gpg --imports arquivo                # importa todas as chaves a partir de arquiv
$ gpg --send-keys ID_usr               # envia a chave de ID_usr para o servidor de
$ gpg --recv-keys ID_usr               # recebe a chave de ID_usr a partir do servi
$ gpg --list-keys ID_usr               # lista as chaves de ID_usr
$ gpg --list-sigs ID_usr               # lista assinatura de ID_usr
$ gpg --check-sigs ID_usr              # confere assinatura de ID_usr
$ gpg --fingerprint ID_usr             # confere impressão digital de ID_usr
$ gpg --list-sigs | grep ’^sig’ | grep ’[User id not found]’ \
  | awk ’{print $2}’ | sort -u | xargs gpg --recv-keys # lista chaves desconhecida
  # atualiza chaves para todas assinaturas desconhecidas.
```

Códigos de confiabilidade:

```
-         Nenhum proprietário confiável associado / ainda não calculado.
e         Cálculo de confiabilidade falhou.
q         Informação insuficiente para cálculo.
n         Nunca confie nesta chave.
m         Marginalmente confiável.
f         Totalmente confiável.
u         Ultimamente confiável.
```

O que se segue enviará minha chave *"A8061F32"* para múltiplos servidores de chaves:

```
$ for xx in us es cz de dk uk ch net.uk earth.net.uk; \
$ do gpg --keyserver wwwkeys.$xx.pgp.net  --send-keys A8061F32; done
```

14.4 Usando GnuPG com aplicações

14.4.1 Usando GnuPG com Mutt

Adicione o seguinte ao `~/.muttrc` para evitar que um GnuPG lento inicie automaticamente, permitindo que seja usado teclando '`S`' no índice de menu.

```
macro index S ":toggle pgp_verify_sig\n"
set pgp_verify_sig=no
```

14.4.2 Usando GnuPG com o Vim

Adicione o conteúdo de `_vimrc` obtido a partir do subdiretório de exemplos (`http://www.debian.org/doc/manuals/debian-reference/examples/`) ao `~/.vimrc` para rodar GnuPG transparentemente.

Capítulo 15

Suporte para o Debian

Os recursos seguintes fornecem ajuda, conselhos, e suporte para o Debian. Tente o máximo possível usar os recursos de auto-ajuda antes de sair chorando nas listas de discussão. :)

Note que você pode acessar muita documentação em seu sistema usando um navegador WWW, através dos comandos `dwww` ou `dhelp`, encontrados em seus respectivos pacotes.

15.1 Referências

As referências seguintes estão disponíveis para o Debian e o Linux em geral. Se seus conteúdos conflitarem uns com os outros, confie sempre mais nas fontes de informação primárias que nas secundárias como esse documento.

- Manual de Instalação (primária)
  - Leia antes da instalação e atualização.
  - Rede: `http://www.debian.org/releases/stable/installmanual`
  - Rede: `http://www.debian.org/releases/testing/installmanual` (em construção, algumas vezes pode não existir)
  - Pacote: `Not available in install-doc: Bug#155374`
  - Arquivo: `DebianCDunder/doc/`

- Notas de Lançamento (primária)
  - Uma leitura obrigatória antes da instalação e atualização mesmo para os experientes.
  - Rede: `http://www.debian.org/releases/stable/releasenotes`
  - Rede: `http://www.debian.org/releases/testing/releasenotes` (em construção, algumas vezes pode não existir)
  - Pacote: `Not available in install-doc: Bug#155374`
  - Arquivo: `DebianCDunder/doc/`

- FAQ (secundária)

  - Questões feitas freqüentemente
  - Rede: `http://www.debian.org/doc/manuals/debian-faq/`
  - Pacote: `doc-debian`
  - Arquivo: `/usr/share/doc/debian/FAQ/index.html`

- Referência Debian (secundária)
  - Manual do usuário pós-instalação muito compreensível
  - Rede: `http://www.debian.org/doc/manuals/debian-reference/`
  - Pacote: `debian-reference-pt-br`
  - Arquivo: `/usr/share/doc/Debian/reference/`

- APT HOWTO (secundária)
  - Guia do usuário detalhado para o gerenciamento de pacotes do Debian. (Woody)
  - Rede: `http://www.debian.org/doc/manuals/apt-howto/`
  - Pacote: `apt-howto`
  - Arquivo: `/usr/share/doc/Debian/apt-howto/`

- Manual de Segurança Debian (secundária)
  - Guia do usuário detalhado sobre segurança da instalação padrão do Debian. (Woody)
  - Rede: `http://www.debian.org/doc/manuals/securing-debian-howto/`
  - Pacote: `harden-doc`
  - Arquivo: `/usr/share/doc/harden-doc/html/securing-debian-howto/`

- Documentação do `dselect` para Iniciantes (secundária)
  - Tutorial sobre o `dselect`
  - Rede: `http://www.debian.org/releases/woody/i386/dselect-beginner`
  - Pacote: `Not available in install-doc: Bug#155374`
  - Arquivo: `DebianCDunder/doc/`

- Manual de Políticas Debian (primária)
  - Espinha dorsal técnica do Debian.
  - Rede: `http://www.debian.org/doc/debian-policy/`
  - Pacote: `debian-policy`
  - Arquivo: `/usr/share/doc/debian-policy/`

- Referência dos Desenvolvedores Debian (primária)
  - Conhecimento básico para desenvolvedores.
  - O resto de nós também deve lê-lo uma vez.
  - Rede: `http://www.debian.org/doc/manuals/developers-reference/`
  - Pacote: `developers-reference`
  - Arquivo: `/usr/share/doc/developers-reference/`

- Guia dos Novos Desenvolvedores Debian (primária)
  - Guia prático para desenvolvedores.
  - Tutoriais de empacotamento para o resto de nós.

  - Rede: `http://www.debian.org/doc/manuals/maint-guide/`
  - Pacote: `maint-guide`
  - Arquivo: `/usr/share/doc/maint-guide/`

- Manual de Empacotamento (Potato)
  - Pacote `packaging-manual` do Potato. (Movido para o apêndice da *Referência dos Desenvolvedores*.)

- Páginas de manual estilo Unix (primária)
  - `man` *`nome-do-pacote`*

- Páginas info estilo GNU (primária)
  - `info` *`nome-do-pacote`*

- Documentos específicos do pacote (primária)
  - Encontre-os sob o `/usr/share/doc/`*`nome-do-pacote`*

- LDP: Linux Documentation Project (Projeto de Documentação Linux) (secundária)
  - HOWTOs gerais do Linux e mini-HOWTOs
  - Rede: `http://www.tldp.org/`
  - Pacote: `doc-linux-text`
  - Arquivo: `/usr/share/doc/HOWTO/`

- DDP: Debian Documentation Project (Projeto de Documentação Debian) (secundária)
  - Manuais específicos do Debian
  - Rede: `http://www.debian.org/doc/`

- Canto dos Desenvolvedores Debian (secundária)
  - Informações chave para os desenvolvedores Debian
  - Esclarecedor para os usuários finais
  - Rede: `http://www.debian.org/devel/`

- Código fonte (absolutamente primária)
  - Ninguém pode contestar isso :-)
  - Baixe o código fonte seguindo 'O código-fonte' on page 11

As referências seguintes estão disponíveis para o Unix em geral. Por favor, note que há algumas pequenas diferenças entre os sistemas Unix. Os nomes de dispositivos e os métodos de init merecem atenção extra.

- *The UNIX Programming Environment* (O Ambiente de Programação Unix)
  - O livro a ser lido para aprender como o UNIX funciona.
  - De B. W. Kernighan e R. Pike
  - Publicado pela Princeton Hall Software Series

- *The C Programming Language* (A Linguagem de Programação C) (segunda edição)
  - O livro a ser lido para aprender ANSI C.

  - De B. W. Kernighan e D. M. Ritchie
  - Publicado pela Princeton Hall Software Series

- *UNIX Power Tools* (Ferramentas Poderosas UNIX)
  - O livro a ser lido para aprender dicas do Unix.
  - De Jerry Peek, Tim O'Reilly, e Mike Loukides
  - Publicado pela O'Reilly and Associates

- *Essential System Administration* (Essencial da Administração de Sistema) (segunda edição)
  - O livro a ser lido para aprender administração de sistemas Unix para vários sabores de Unix.
  - De Aeleen Frisch
  - Publicado pela O'Reilly and Associates

- Bell Labs: Computing Sciences Research (Bell Labs: Pesquisa de Ciências da Computação)
  - Um rico arquivo da história do Unix
  - Principal: `http://cm.bell-labs.com/cm/cs/`
  - Relatórios técnicos selecionados: `http://cm.bell-labs.com/cm/cs/cstr.html`
  - Alguns artigos: `http://cm.bell-labs.com/cm/cs/papers.html`

- Recursos de suporte Linux gerais na rede
  - Debian Planet (`http://www.debianplanet.org/`)
  - debianHELP (`http://www.debianhelp.org/`)
  - Linux.com (`http://linux.com/`)
  - A Página do Linux em Linux Online (`http://www.linux.org/`)
  - Red Hat (empresa de Linux comercial) (`http://www.redhat.com/`) (RPM, Sys-V init)
  - SuSE, Inc. (empresa de Linux comercial) (`http://www.suse.de/`) (RPM, init Sys-V)
  - Slackware (`http://www.slackware.com/`) (TGZ, init estilo BSD)

- Recursos guia gerais de Unix na rede
  - Um curso introdutório de UNIX da Ohio State University (`http://www-wks.acs.ohio-state.edu/unix_course/unix.html`)
  - UNIXhelp da University of Edinburgh (`http://unixhelp.ed.ac.uk/`)
  - Unix / Informação de Programação (`http://arioch.unomaha.edu/~jclark/#info`)
  - FAQ comp.unix.questions (`http://www.faqs.org/faqs/unix-faq/faq/`)
  - FAQ comp.unix.user-friendly (`http://www.camelcity.com/~noel/usenet/cuuf-FAQ.htm`)
  - Documentação FreeBSD (`http://www.freebsd.org/docs.html`)
  - O Manual FreeBSD (`http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/index.html`)
  - GUIA UNIX (`http://ieee.uow.edu.au/documents/`)
  - O Patrimônio da Sociedade Unix (`http://www.tuhs.org/`)

- Páginas de projetos de Software Livre
    - Projeto GNU (`http://www.gnu.org/`)
    - O Projeto de Documentação Linux (`http://www.tldp.org/`)
    - Os Arquivos do Kernel Linux (`http://www.linux.org/`)
    - O Projeto XFree86, Inc (`http://www.xfree86.org/`)
    - GNOME (`http://www.gnome.org/`)
    - K Desktop Environment (`http://www.kde.org/`)
    - Software GNU no Red Hat (`http://sources.redhat.com/`)
    - Mozilla (`http://www.mozilla.org`)
    - FreeBSD (`http://www.freebsd.org/`)
    - OpenBSD (`http://www.openbsd.org/`)
    - NetBSD (`http://www.netbsd.org/`)

15.2 Descobrindo o significado de uma palavra

Muitas palavras usadas no Debian são um jargão codificado ou acrônimos. Isso resolverá muitas questões:

```
$ dict ponha-uma-palavra-estranha-aqui
```

15.3 Descobrindo a popularidade de um pacote Debian

Há muitos pacotes no Debian e às vezes é difícil saber qual tentar primeiro. Veja os Resultados do torneio de popularidade Debian (`http://popcon.debian.org/`) para ter uma idéia clara do que os outros estão usando. Instale também o pacote `popularity-contest` para contribuir.

15.4 O sistema de gerenciamento de bugs Debian

A distribuição Debian tem um sistema de gerenciamento de bugs (BTS) (`http://bugs.debian.org/`) que dá os detalhes dos bugs relatados por usuários e desenvolvedores. É dado um número a cada bug, e ele é mantido ativo até que seja marcado como resolvido.

Você deve verificar se seu relatório de bug já foi feito por mais alguém antes de enviá-lo. As listas dos bugs pendentes estão disponíveis na World Wide Web (`http://bugs.debian.org/`) e também aqui (`http://www.debian.org/Bugs/Access`). Veja também 'Verificar bugs no Debian e procurar ajuda' on page 88.

Podem existir muitos relatórios de bugs críticos para o lançamento marcados com **FTBFS**. Isso significa "Fails To Build From Source" (Falha ao Compilar a partir dos Fontes).

As instruções para relatar um bug são dadas em `http://www.debian.org/Bugs/Reporting`.

15.5 Listas de discussão

Leia pelo menos a ″debian-devel-announce“ (inglês, de somente leitura e tráfego baixo) para ficar atualizado sobre o Debian.

As listas de discussão mais interessantes para os usuários Debian são a ″debian-user“ (inglês, aberta e alto tráfego) e outras listas ″debian-user-*idioma*“ (para outros idiomas).

Se quiser mais informações sobre essas listas e detalhes sobre como inscrever-se, veja `http://lists.debian.org/`. Por favor, procure nos arquivos as respostas para a sua questão antes de enviar mensagens e também adira à etiqueta padrão da lista.

Se você não não desejar receber respostas CC às suas mensagens enviadas à lista de discussão, use o cabeçalho `Mail-Followup-To:` que é uma medida muito efetiva. Essa a convenção informal das listas de discussão como explicado em `http://cr.yp.to/proto/replyto.html`.

15.6 Internet Relay Chat (IRC)

O IRC (Internet Relay Chat) é uma forma de conversar com pessoas de todo o mundo em tempo real. Os canais de IRC dedicados ao Debian podem ser encontrados na rede de IRC freenode (`http://www.freenode.info/`). Para conectar, você precisa de um cliente IRC. Alguns dos clientes mais populares são XChat, BitchX, ircII, irssi, epic4 e KSirc, todos os quais foram empacotados para o Debian. Tendo instalado o cliente, você precisa fazê-lo conectar no servidor. Em muitos clientes, você pode fazer isso digitando:

```
/server irc.debian.org
```

Uma vez conectado, divirta-se no canal #debian digitando

```
/join #debian
```

Para deixar o canal #debian digite

```
/part #debian
```

Você pode sair do cliente irc digitando

```
/quit
```

Para enviar uma mensagem particular ″Olá Sr. Foo“ para *foo* digite

```
/msg foo Olá Sr. Foo
```

Note que qualquer coisa que você digitar sem uma `/` precedendo é enviado para o canal como uma mensagem.

Nota: clientes como o XChat geralmente têm uma interface gráfica diferente para entrar em servidores/canais.

15.7 Mecanismos de busca

Há muitos mecanismos de busca que fornecem documentação relacionada ao Debian:

- Página de busca WWW Debian (`http://search.debian.org/`).
- Google (`http://www.google.com/`): inclua "site:debian.org" como termo de busca.
- Google Groups (`http://groups.google.com/`): um mecanismo de busca para grupos de notícias. Inclua "group:linux.debian.*" como termo de busca.
- AltaVista (`http://www.altavista.com/`)

Por exemplo, procurando pelo termo "cgi-perl", obtém-se uma explicação mais detalhada desse pacote que o campo de descrição breve em seu arquivo control. Veja 'Verificar bugs no Debian e procurar ajuda' on page 88 para obter informações relacionadas.

15.8 Sítios Web

Esses são algumas URLs aleatórias que eu reuni para problemas específicos.

- IBM developerWorks: Linux (`http://www.ibm.com/developerworks/linux/`)
- Últimos pacotes de Adrian Bunk (back port para stable) (`http://www.fs.tum.de/~bunk/`)
- Linux em Laptops (`http://www.linux-laptop.net/`)
- Xterm FAQ (`http://dickey.his.com/xterm/xterm.faq.html`)
- mini-HOWTO de Sistema de Arquivos EXT3 (`http://www.zip.com.au/~akpm/linux/ext3/ext3-usage.html`)
- Suporte a arquivos grandes no Linux (`http://www.suse.de/~aj/linux_lfs.html`)
- Gerenciadores de Janela para o X (`http://www.xwinman.org`)
- Projeto USB Linux (`http://www.linux-usb.org/`)

- Páginas da SuSE para CJK (`http://www.suse.de/~mfabian/suse-cjk/suse-cjk.html`)
- LNX-BBC (Projeto de CD inicializável do tamanho de um cartão de visitas) (`http://www.lnx-bbc.org/`)
- Linux info de Karsten Self (particionamento, cópia de segurança, navegadores...) (`http://kmself.home.netcom.com/Linux/index.html`)
- Backup info HOWTO de Alvin Oga (`http://www.Linux-Backup.net/`)
- Security info HOWTO de Alvin Oga (`http://www.Linux-Sec.net/`)
- Várias fontes NÃO OFICIAIS para o APT (`http://www.apt-get.org/`)
- Configuração de Ethernet para Laptop (`http://www.orthogony.com/gjw/lap/lap-ether-intro.html`)

Apêndice A

Apêndice

A.1 Autores

O Referência Debian foi iniciado por Osamu Aoki `<osamu\#at\#debian.org>` como um registro pessoal de instalação que foi eventualmente chamado ″Referência Rápida″. Muitas partes foram extraídas dos arquivos da lista de discussão ″debian-user″. Além disso, também foram usados o ″Manual de Instalação Debian″ e ″Notas de Lançamento Debian″

Seguindo uma sugestão de Josip Rodin, que é muito ativo no Projeto de Documentação Debian (`http://www.debian.org/doc/ddp`) (DDP) e é o atual mantenedor da ″Debian FAQ″, esse documento foi renomeado como ″Referência Debian″ e foi fundido com vários capítulos da ″Debian FAQ″ com conteúdo de referência. Então a ″Referência Rápida Debian″ é formada de resumos.

Esse documento foi editado, traduzido, e ampliado pelos seguintes membros do time QREF:

- Originais em inglês do ″Referência Rápida. . . ″ original
  - Osamu Aoki `<osamu\#at\#debian.org>` (líder: todos os conteúdos)
- Revisão do inglês e contribuição adicional
  - David Sewell `<dsewell\#at\#virginia.edu>` (líder: estilo en)
  - Thomas Hood `<jdthood\#at\#yahoo.co.uk>` (relacionado a rede)
  - Brian Nelson `<nelson\#at\#bignachos.com>` (especialmente relacionado ao X)
  - Jan Michael C Alonzo `<jmalonzo\#at\#spaceants.net>`
  - Daniel Webb `<webb\#at\#robust.colorado.edu>`
  - Retorno de todos os tradutores
- Tradução para o francês
  - Guillaume Erbs `<gerbs\#at\#free.fr>` (líder: fr)
  - Rénald Casagraude `<rcasagraude\#at\#interfaces.fr>`
  - Jean-Pierre Delange `<adeimantos\#at\#free.fr>`
  - Daniel Desages `<daniel\#at\#desages.com>`
- Tradução para o italiano
  - Davide Di Lazzaro `<mc0315\#at\#mclink.it>` (líder: it)
- Tradução para o português (Brasil)

  - Paulo Rogério Ormenese `<pormenese\#at\#uol.com.br>` (líder: pt-br)
  - Andre Luis Lopes `<andrelop\#at\#ig.com.br>`
  - Marcio Roberto Teixeira `<marciotex\#at\#pop.com.br>`
  - Rildo Taveira de Oliveira `<to_rei\#at\#yahoo.com>`
  - Raphael Bittencourt Simoes Costa `<raphael-bsc\#at\#bol.com.br>`
  - Gustavo Noronha Silva `<kov\#at\#debian.org>` (coordinator)
- Tradução para o espanhol
  - Walter Echarri `<wecharri\#at\#infovia.com.ar>` (líder: es)
  - José Carreiro `<ffx\#at\#urbanet.ch>`
- Tradução para o alemão
  - Jens Seidel `<tux-master\#at\#web.de>` (líder: de)
  - Willi Dyck `<wdyck\#at\#gmx.net>`
  - Stefan Schröder `<stefan\#at\#fkp.uni-hannover.de>`
  - Agon S. Buchholz `<asb\#at\#kefk.net>`
- Tradução para o polonês—os seguintes membros do PDDP (`http://debian.linux.org.pl`):
  - Marcin Andruszkiewicz
  - Mariusz Centka `<mariusz.centka\#at\#debian.linux.org.pl>`
  - Bartosz Feński `<fenio\#at\#debian.linux.org.pl>` (líder: pl)
  - Radosław Grzanka `<radekg\#at\#debian.linux.org.pl>`
  - Bartosz 'Xebord' Janowski
  - Jacek Lachowicz
  - Rafał Michaluk
  - Leonard Milcin, Jr.
  - Tomasz Z. Napierała `<zen\#at\#debian.linux.org.pl>`
  - Oskar Ostafin `<cx\#at\#debian.linux.org.pl>`
  - Tomasz Piękoś
  - Jacek Politowski
  - Mateusz Prichacz `<mateusz\#at\#debian.linux.org.pl>`
  - Marcin Rogowski
  - Paweł Różański
  - Mariusz Strzelecki
  - Krzysztof Ścierski
  - Przemysław Adam Śmiejek `<tristan\#at\#debian.linux.org.pl>`
  - Mateusz Tryka `<uszek\#at\#debian.linux.org.pl>`
  - Cezary Uchto
  - Krzysztof Witkowski `<tjup\#at\#debian.linux.org.pl>`
  - Bartosz Zapałowski `<zapal\#at\#debian.linux.org.pl>`
- Tradução para o chinês (simplificado)
  - Hao "Lyoo" LIU `<iamlyoo\#at\#163.net>`
  - Ming Hua `<minghua\#at\#rice.edu>`
  - Xiao Sheng Wen `<atzlinux\#at\#163.com>` (l der: zh-cn)
  - Haifeng Chen `<optical.dlz\#at\#gmail.com>`
  - Xie Yanbo `<xieyanbo\#at\#gmail.com>`
  - easthero `<easthero\#at\#gmail.com>`
- Tradução para o chinês (tradicional)

  - Asho Yeh <`asho\#at\#debian.org.tw`> (líder: zh-tw)
  - Tang Wei Ching <`wctang\#at\#csie.nctu.edu.tw`> (ex-líder: zh-tw)
- Tradução para o japonês
  - Shinichi Tsunoda <`tsuno\#at\#ngy.1st.ne.jp`> (líder: ja)
  - Osamu Aoki <`osamu\#at\#debian.org`>

QREF é uma abreviação do título do documento original, "Quick Reference. . . " e também é o nome do projeto em qref.sourceforge.net.

Muitas páginas de manual e páginas info no sistema Debian foram usadas como as referências primárias para escrever este documento. Para a extensão, Osamu Aoki cosiderou dentro do bom uso de citações, muitas partes delas, especialmente definições de comandos, que foram usadas como partes de frases depois de esforços editoriais cuidadosos para adequá-las ao estilo e objetivo deste documento.

Muitas partes de 'Fundamentos Debian' on page 5 vêm originalmente da "Debian FAQ" (março de 2002):

- 5. Os arquivos de FTP do Debian: `ftparchives.sgml` (o capítulo todo)
- 6. O básico do sistema de gerenciamento de pacotes do Debian: `pkg_basics.sgml` (o capítulo todo)
- 7. As ferramentas de gerenciamento de pacotes do Debian: `pkgtools.sgml` (o capítulo todo)
- 8. Mantendo seu sistema Debian atualizado: `uptodate.sgml` (capítulo todo)
- 9. O Debian e o kernel: `kernel.sgml` (o capítulo todo)
- 10. Personalizando seu sistema Debian GNU/Linux: `customizing.sgml` (parte do capítulo)

Essas seções da "Debian FAQ" foram incluídas nesse documento depois de uma grande reorganização para refletir as mudanças no sistema Debian. Ambos os documentos são atualizados simultaneamente agora.

A "Debian FAQ" original foi feita e mantida por J.H.M. Dassen (Ray) e Chuck Stickelman. Os autores da "Debian FAQ" reescrita são Susan G. Kleinmann e Sven Rudolph. Depois deles, a "Debian FAQ" foi mantida por Santiago Vila. O atual mantenedor é Josip Rodin.

Partes das informações da "Debian FAQ" vieram de:

- O anúncio do lançamento do Debian-1.1, de Bruce Perens (`http://www.perens.com/`).
- A Linux FAQ, de Ian Jackson (`http://www.chiark.greenend.org.uk/~ijackson/`).
- Arquivos das listas de discussão Debian (`http://lists.debian.org/`),
- O manual dos programadores do dpkg e o manual de Políticas Debian (veja 'Referências' on page 249)
- Muitos desenvolvedores, voluntários, testadores beta, e
- as memórias faiscantes de seus autores. :-)

Algumas partes da seção "Tutorial" foram derivadas de

- "Debian Tutorial" de Havoc Pennington, Oliver Elphick, Ole Tetlie, James Treacy, Craig Sawyer, and Ivan E. Moore II. (Esse documento foi derivado do "Linux User's Guide" de Larry Greenfield.)

- ″Debian GNU/Linux: Guide to Installation and Usage″ de John Goerzen e Ossama Othman.

Os autores gostariam de agradecer a todos aqueles que ajudaram a tornar esse documento possível.

A.2 Garantias

Como eu não sou um expert, não pretendo ser totalmente compreensível quanto ao Debian ou ao Linux em geral. As considerações de segurança que faço devem ser aplicáveis apenas ao uso doméstico.

Este documento não substitui nenhum guia oficial.

Não se oferecem garantias de nenhum tipo. Todas as marcas registradas são de propriedade de seus respectivos donos.

A.3 Retorno

Comentários e adendos a esse documento são sempre bem-vindos. Por favor, envie uma mensagem para o sistema BTS Debian (`http://bugs.debian.org/`) relacionada ao pacote `debian-reference` ou para seus respectivos pacotes de traduções. O uso do `reportbug` facilita o envio de um relatório de bug completo. Você pode ainda enviar uma mensagem em inglês para Osamu Aoki (`http://people.debian.org/~osamu/`) em `<osamu\#at\#debian.org>` ou para cada tradutor em seus respectivos idiomas.

Apesar de ter vivido nos Estados Unidos, não sou usuário nativo do inglês. Quaisquer correções gramaticais são bem-vindas.

O melhor retorno é um patch para a versão SGML, mas um patch para a versão texto também é bem-vindo. Veja 'Documento Oficial' on page 1 para ver o sítio oficial do documento.

Os arquivos SGML originais usados para criar este documento também estão disponíveis no CVS em: `:pserver:anonymous@cvs.sf.net/cvsroot/qref` ou `http://qref.sourceforge.net/Debian/qref.tar.gz`.

A.4 Formato do documento

Este documento foi escrito usando o DTD SGML DebianDoc (reescrito a partir do SGML LinuxDoc). O sistema SGML DebianDoc possibilita a criação de arquivos em vários formatos a partir de uma fonte, por exemplo, este documento pode ser visto em HTML, texto plano, TeX DVI, PostScript, PDF e GNU info.

Os utilitários de conversão para o SGML DebianDoc estão disponíveis no pacote Debian `debiandoc-sgml`.

A.5 O labirinto do Debian

O sistema Linux é uma plataforma para computadores em rede muito poderoso. Entretanto, o aprendizado do uso de todas as suas capacidades não é fácil. A configuração de uma impressora é um bom exemplo.

Há um mapa completo e detalhado chamado ″CÓDIGO FONTE‴. Ele é muito preciso, mas é muito difícil de entender. Também há referências chamadas HOWTO e mini-HOWTO. Elas são mais fáceis de entender mas tendem a dar muitos detalhes e perder o foco. Algumas vezes, tenho problemas para encontrar a seção correta em um longo HOWTO quando preciso apenas chamar uns poucos comandos.

Para navegar nesse labirinto da configuração de um sistema Linux, comecei a registrar lembretes simples em arquivo de formato texto para minha referência rápida. Essa lista de lembretes aumentou e eu aprendi o debiandoc nesse meio tempo. O produto é essa *Referência Debian*.

A.6 As citações do Debian

Aqui estão algumas citações interessantes da lista de discussão do Debian.

- ″This is Unix. It gives you enough rope to hang yourself.‴ — Miquel van Smoorenburg `<miquels@cistron.nl>` (″Isto é Unix. Ele lhe dá corda suficiente para enforcar a si mesmo.‴)
- ″Unix **IS** user friendly... It's just selective about who its friends are.‴ — Tollef Fog Heen `<tollef@add.no>` (″Unix É amigável... Ele só é seletivo quanto a quem são seus amigos.‴)