

debian

Referência Debian

Osamu Aoki

Copyright © 2013-2018 Osamu Aoki

Esta Referência Debian (versão 2.152) (2026-10-10 14:56:25 UTC) pretende fornecer uma visão geral do sistema Debian como um guia do utilizador pós-instalação. Cobre muitos aspetos da administração do sistema através de exemplos shell-command para não programadores.

Índice

1 Manuais de GNU/Linux **1**

1.1 Básico da consola . . . 1

1.1.1 A linha de comandos da shell . . . 1

1.1.2 A linha de comandos na GUI . . . 2

1.1.3 A conta root . . . 2

1.1.4 A linha de comandos shell do root . . . 3

1.1.5 GUI de ferramentas de administração do sistema . . . 3

1.1.6 Consolas virtuais . . . 3

1.1.7 Como abandonar a linha de comandos . . . 4

1.1.8 Como desligar o sistema . . . 4

1.1.9 Recuperar uma consola sã . . . 4

1.1.10 Sugestões de pacotes adicionais para o novato . . . 4

1.1.11 Uma conta de utilizador extra . . . 5

1.1.12 Configuração do sudo . . . 5

1.1.13 Hora de brincar . . . 6

1.2 Sistema de ficheiros tipo Unix . . . 6

1.2.1 Noções básicas de ficheiros Unix . . . 7

1.2.2 Internos do sistema de ficheiros . . . 8

1.2.3 Permissões do sistema de ficheiros . . . 8

1.2.4 Controlo de permissões para ficheiros acabados de criar: umask . . . 11

1.2.5 Permissões para grupos de utilizadores (group) . . . 12

1.2.6 Marcas temporais (Timestamps) . . . 13

1.2.7 Links (ligações) . . . 14

1.2.8 Pipes com nome (FIFOs) . . . 15

1.2.9 Sockets . . . 16

1.2.10 Ficheiros de dispositivo . . . 16

1.2.11 Ficheiros de dispositivos especiais . . . 17

1.2.12 procfs e sysfs . . . 17

1.2.13 tmpfs . . . 18

1.3 Midnight Commander (MC) . . . 18

1.3.1 Personalização do MC . . . 19
1.3.2 Iniciar o MC . . . 19
1.3.3 Gestor de ficheiros no MC . . . 19
1.3.4 Truques de linha de comandos no MC . . . 20
1.3.5 O editor interno em MC . . . 20
1.3.6 O visualizador interno no MC . . . 20
1.3.7 Funcionalidades de auto-arranque do MC . . . 21
1.3.8 Sistema de ficheiros virtual do MC . . . 21
1.4 O ambiente de trabalho estilo Unix básico . . . 21
1.4.1 A shell de login . . . 21
1.4.2 Personalizar bash . . . 21
1.4.3 Teclas especiais . . . 23
1.4.4 Operações com o rato . . . 23
1.4.5 O pager . . . 24
1.4.6 O editor de texto . . . 24
1.4.7 Definir um editor de texto predefinido . . . 25
1.4.8 Utilizando o vim . . . 25
1.4.9 Gravar as atividades da shell . . . 26
1.4.10 Comandos básicos de Unix . . . 26
1.5 O comando simples da shell . . . 28
1.5.1 Execução do comando e variável de ambiente . . . 28
1.5.2 A variável "$LANG" . . . 29
1.5.3 A variável "$PATH" . . . 30
1.5.4 A variável "$HOME" . . . 30
1.5.5 Opções da linha de comandos . . . 31
1.5.6 Glob da shell . . . 31
1.5.7 Valor de retorno do comando . . . 32
1.5.8 Sequências de comandos típicas e redireccionamento da shell . . . 32
1.5.9 Comando alias . . . 34
1.6 Processamento de texto estilo Unix . . . 35
1.6.1 Ferramentas de texto de Unix . . . 35
1.6.2 Expressões regulares . . . 36
1.6.3 Expressões de substituição . . . 37
1.6.4 Substituição global com expressões regulares . . . 37
1.6.5 Extrair dados de tabela de ficheiro de texto . . . 38
1.6.6 Trechos de script para canalizar comandos em pipe . . . 40

2 Gestão de pacotes Debian **41**

2.1 Pré-requisitos da gestão de pacotes Debian . . . 41
2.1.1 Sistema de gestão de pacotes Debian . . . 41
2.1.2 Configuração de pacotes . . . 42
2.1.3 Precauções básicas . . . 42
2.1.4 A vida com atualizações eternas . . . 43
2.1.5 Básico do arquivos Debian . . . 44
2.1.6 Debian é 100% software livre . . . 48
2.1.7 Dependências de pacote . . . 49
2.1.8 O fluxo de eventos da gestão de pacotes . . . 50
2.1.9 Primeira resposta a problemas com a gestão de pacotes . . . 51
2.1.10 Como escolher os pacotes Debian . . . 52
2.1.11 Como lidar com requisitos contraditórios . . . 52
2.2 Operações básicas de gestão de pacotes . . . 53
2.2.1 `apt` vs. `apt-get` / `apt-cache` vs. `aptitude` . . . 53
2.2.2 Operações básicas de gestão de pacotes com a linha de comandos . . . 54
2.2.3 Uso interativo do aptitude . . . 54
2.2.4 Teclas de atalho do aptitude . . . 56
2.2.5 Vistas de pacote no aptitude . . . 56
2.2.6 Opções do método de pesquisa com o aptitude . . . 57
2.2.7 A fórmula regex do aptitude . . . 58
2.2.8 Resolução de dependências do aptitude . . . 60
2.2.9 Relatórios (logs) de atividade de pacotes . . . 60
2.3 Exemplos de operações do aptitude . . . 60
2.3.1 Procurar pacotes interessantes . . . 60
2.3.2 Listagem de pacotes com correspondência por expressão regular nos nomes de pacotes . . . 60
2.3.3 Explorar com a correspondência de expressão regular . . . 60
2.3.4 Purgar pacotes removidos definitivamente . . . 61
2.3.5 Acertar o estado auto/manual de instalação . . . 61
2.3.6 atualização total ao sistema . . . 62
2.4 Operações de gestão avançada de pacotes . . . 63
2.4.1 Operações de gestão avançada de pacotes com linha de comandos . . . 63
2.4.2 Verificação dos ficheiros pacotes instalados . . . 65
2.4.3 Salvaguardar para problemas de pacotes . . . 65
2.4.4 Procurar nos meta-dados do pacote . . . 65
2.5 Os interiores da gestão de pacotes Debian . . . 65
2.5.1 Meta dados do arquivo . . . 65
2.5.2 Ficheiro ”Release” de nível de topo e autenticidade: . . . 66
2.5.3 Ficheiros ”Release” do nível de arquivo . . . 67

2.5.4 Obter os meta dados do pacote . . . 68
2.5.5 O estado dos pacote para o APT . . . 68
2.5.6 O estado de pacotes para o aptitude . . . 68
2.5.7 Copias locais dos pacotes obtidos . . . 68
2.5.8 Nomes de ficheiros de pacotes Debian . . . 69
2.5.9 O comando dpkg . . . 69
2.5.10 O comando update-alternatives . . . 70
2.5.11 O comando dpkg-statoverride . . . 71
2.5.12 O comando dpkg-divert . . . 71
2.6 Recuperação de um sistema danificado . . . 71
2.6.1 Incompatibilidade com configurações antigas de utilizador . . . 71
2.6.2 Erros de armazenamento em cache dos dados do pacote . . . 72
2.6.3 Recuperação com o comando dpkg . . . 72
2.6.4 Falha na instalação devido a dependências em falta . . . 72
2.6.5 Pacotes diferentes com ficheiros sobrepostos . . . 73
2.6.6 Corrigir script problemático de pacote . . . 73
2.6.7 Recuperar dados de seleção de pacotes . . . 73
2.7 Dicas para a gestão de pacotes . . . 74
2.7.1 Quem fez o upload do pacote? . . . 74
2.7.2 Limitar a largura de banda de descarga para o APT . . . 74
2.7.3 Descarga e atualização automática de pacotes . . . 74
2.7.4 Atualizações e Backports . . . 75
2.7.5 Arquivos de pacotes externos . . . 75
2.7.6 Pacotes de fontes mistas de arquivos sem **apt-pinning** . . . 76
2.7.7 Ajustar a versão candidata com o **apt-pinning** . . . 77
2.7.8 Bloquear pacotes instalados por ”Recomendados” . . . 78
2.7.9 Acompanhar `testing` com alguns pacotes de `unstable` . . . 78
2.7.10 Acompanhar `unstable` com alguns pacotes de `experimental` . . . 80
2.7.11 Downgrade de emergência . . . 80
2.7.12 O pacote equivs . . . 81
2.7.13 Portar um pacote ao sistema stable . . . 82
2.7.14 Servidor proxy para o APT . . . 82
2.7.15 Mais leituras sobre a gestão de pacotes . . . 83

3 A inicialização do sistema 84
3.1 Uma visão geral do processo de arranque . . . 84
3.1.1 Estágio 1: a UEFI . . . 84
3.1.2 Estágio 2: o gestor de arranque . . . 85
3.1.3 Estágio 3: o mini-sistema Debian . . . 86
3.1.4 Estágio 4: o sistema Debian normal . . . 87
3.2 O sistema de salvamento . . . 87
3.2.1 Sistema de salvamento GRUB UEFI em USB . . . 88
3.2.2 Sistema de salvamento Linux live em USB . . . 89
3.2.3 Sistema de salvamento Linux live a partir do GRUB . . . 90
3.3 Systemd . . . 90
3.3.1 init do Systemd . . . 90
3.3.2 Inicio de sessão Systemd . . . 91
3.4 A mensagem do kernel . . . 92
3.5 A mensagem do sistema . . . 92
3.6 Gestão do sistema . . . 93
3.7 Outros monitores de sistema . . . 93
3.8 Configuração do sistema . . . 93
3.8.1 O nome da máquina . . . 93
3.8.2 O sistema de ficheiros . . . 93
3.8.3 Inicialização da interface de rede . . . 95
3.8.4 Inicialização do sistema de nuvem . . . 95
3.8.5 Exemplo de personalização para ajustar o serviço sshd . . . 95
3.9 O sistema udev . . . 96
3.10 A inicialização de módulos do kernel . . . 96

4 Autenticação e controlos de acesso 98
4.1 Autenticação normal de Unix . . . 98
4.2 Gerir informação de conta e palavra-passe . . . 100
4.3 Boa palavra-passe . . . 100
4.4 Criar palavra-passe encriptada . . . 101
4.5 PAM e NSS . . . 101
4.5.1 Ficheiros de configuração acedidos pelo PAM e NSS . . . 102
4.5.2 O moderno sistema de gestão centralizado . . . 102
4.5.3 ”Porque o su do GNU não suporta o grupo wheel” . . . 103
4.5.4 Regras de palavra-passe rigorosas . . . 103
4.6 Segurança da autenticação . . . 104
4.6.1 Palavra-passe segura na Internet . . . 104
4.6.2 Shell Segura . . . 104

4.6.3 Medidas de segurança extra para a Internet . . . 105
4.6.4 Tornar a palavra-passe do root segura . . . 105
4.7 Outros controles de acesso . . . 106
4.7.1 Listas de controlo de acesso (ACLs) . . . 106
4.7.2 sudo . . . 106
4.7.3 PolicyKit . . . 107
4.7.4 Recadeiair acesso a alguns serviços de servidor . . . 107
4.7.5 Caraterísticas de segurança do Linux . . . 108

5 Configuração de rede 109
5.1 A infra-estrutura de rede básica . . . 109
5.1.1 A resolução de nome de máquina . . . 109
5.1.2 O nome da interface de rede . . . 111
5.1.3 A gama de endereços de rede para a LAN . . . 112
5.1.4 O suporte a dispositivos de rede . . . 112
5.2 A configuração moderna de rede para desktop . . . 112
5.2.1 Ferramentas GUI de configuração de rede . . . 113
5.3 A moderna configuração de rede sem GUI . . . 113
5.4 A configuração moderna de rede para nuvem . . . 114
5.4.1 A configuração moderna de rede para nuvem com DHCP . . . 114
5.4.2 A configuração moderna de rede para nuvem com IP estático . . . 114
5.4.3 A configuração moderna de rede para nuvem com Network Manager . . . 114
5.5 A configuração de rede de baixo nível . . . 115
5.5.1 Comandos iproute2 . . . 115
5.5.2 Operações de rede seguras de baixo nível . . . 115
5.6 Optimização da rede . . . 116
5.6.1 Encontrar o MTU óptimo . . . 116
5.6.2 Optimização WAN TCP . . . 117
5.7 Infraestrutura netfilter . . . 117

6 Aplicações de rede 120
6.1 Navegadores web . . . 120
6.1.1 Falsificação da cadeia User-Agent . . . 121
6.1.2 Extensão do navegador . . . 121
6.2 O sistema de correio electrónico (mail) . . . 121
6.2.1 Noções básicas de mail . . . 121
6.2.2 Limitação do serviço de correio moderno . . . 122
6.2.3 Expectativa histórica do serviço de correio . . . 123
6.2.4 Agente de transporte de mail (MTA) . . . 123

6.2.4.1 A configuração do exim4 . . . 123
6.2.4.2 A configuração do postfix com SASL . . . 125
6.2.4.3 A configuração do endereço de mail . . . 126
6.2.4.4 Operações MTA básicas . . . 127
6.3 O servidor de acesso remoto e utilitários (SSH) . . . 127
6.3.1 Bases do SSH . . . 128
6.3.2 Nome de utilizador no anfitrião remoto . . . 129
6.3.3 Ligar sem palavras-passe remotas . . . 129
6.3.4 Lidar com clientes SSH alienígenas . . . 129
6.3.5 Configurar o ssh-agent . . . 130
6.3.6 Enviar uma mensagem de correio eletrónico a partir de um anfitrião remoto . . . 130
6.3.7 Reencaminhamento de portos para SMTP/POP3 em túnel . . . 130
6.3.8 Como desligar o sistema remoto em SSH . . . 130
6.3.9 Depurar problemas no SSH . . . 131
6.4 O servidor de impressão e utilitários . . . 131
6.5 Outras aplicações de servidor de rede . . . 131
6.6 Outros clientes de aplicação de rede . . . 132
6.7 Os diagnósticos dos daemons do sistema . . . 132

7 Sistema GUI (interface gráfica de utilizador) 134
7.1 Ambiente de trabalho GUI . . . 134
7.2 Protocolo de comunicação GUI . . . 135
7.3 Infraestrutura GUI . . . 136
7.4 Aplicações GUI . . . 137
7.5 Diretórios de utilizador . . . 137
7.6 Fontes . . . 137
7.6.1 Fontes (tipos de letra) básicas . . . 137
7.6.2 Rasterização de tipos de letra . . . 140
7.7 Sandbox . . . 141
7.8 Área de trabalho remota . . . 142
7.9 Ligação ao servidor X . . . 142
7.9.1 Ligação local do servidor X . . . 142
7.9.2 Ligação remota ao servidor X . . . 143
7.9.3 Ligação chroot do servidor X . . . 143
7.10 Área de transferência (Clipboard) . . . 143

8 I18N e L10N **145**

8.1 O locale . . . 145
8.1.1 Fundamentos para o locale UTF-8 . . . 145
8.1.2 A reconfiguração do locale . . . 146
8.1.3 Codificação de nomes de ficheiros . . . 147
8.1.4 Mensagens localizadas e documentação traduzida . . . 147
8.1.5 Efeitos do locale . . . 148
8.2 A entrada do teclado . . . 148
8.2.1 A entrada de teclado para a consola Linux e o X Window . . . 149
8.2.2 A entrada de teclado para o Wayland . . . 149
8.2.3 O suporte a método de entrada com IBus . . . 149
8.2.4 O suporte de método de entrada com Fcitx . . . 150
8.2.5 Um exemplo para Japonês . . . 150
8.3 O ecrã de resultados . . . 152
8.3.1 A configuração do terminal . . . 152
8.3.2 Caracteres Asiáticos de Leste de Altura Ambígua . . . 153

9 Dicas do sistema **154**

9.1 As dicas da consola . . . 154
9.1.1 Gravar as atividades da shell de modo limpo . . . 154
9.1.2 O programa screen . . . 155
9.1.3 Navegando nos diretórios . . . 156
9.1.4 Revestimento da linha de leitura . . . 156
9.1.5 Verificação da árvore de código-fonte . . . 156
9.2 Personalizar o vim . . . 157
9.2.1 Personalizando o vim com recursos internos . . . 157
9.2.2 Personalizando o vim com pacotes externos . . . 159
9.3 Gravação de dados e apresentação . . . 160
9.3.1 O daemon de log . . . 160
9.3.2 Analisador de relatório (Log) . . . 160
9.3.3 Amostragem personalizada de dados em texto . . . 161
9.3.4 Amostragem personalizada de hora e data . . . 161
9.3.5 Echo de shell colorido . . . 162
9.3.6 Comandos coloridos . . . 162
9.3.7 Recordar as atividades do editor para repetições complexas . . . 163
9.3.8 Gravar a imagem gráfica de uma aplicação X . . . 163
9.3.9 Gravar alterações em ficheiros de configuração . . . 163
9.4 Monitorizar, controlar e iniciar as atividades de programas . . . 164
9.4.1 Temporizar um processo . . . 164

9.4.2 A prioridade de agendamento . . . 164
9.4.3 O comando ps . . . 165
9.4.4 O comando top . . . 165
9.4.5 Listar ficheiros abertos por um processo . . . 165
9.4.6 Rastear as atividades de programas . . . 165
9.4.7 Identificação de um processo a usar ficheiros ou sockets . . . 166
9.4.8 Repetir um comando com um intervalo constante . . . 166
9.4.9 Repetir um ciclo de comandos sobre ficheiros . . . 166
9.4.10 Arrancar um programa a partir da GUI . . . 167
9.4.11 Personalizar o programa a ser iniciado . . . 168
9.4.12 Matar um processo . . . 169
9.4.13 Agendar tarefas uma vez . . . 169
9.4.14 Agendar tarefas regularmente . . . 170
9.4.15 Programação de tarefas em eventos . . . 170
9.4.16 Tecla Alt-SysRq . . . 170
9.5 Dicas de manutenção do sistema . . . 171
9.5.1 Quem está no sistema? . . . 171
9.5.2 Avisar todos . . . 172
9.5.3 Identificação do hardware . . . 172
9.5.4 Configuração do hardware . . . 172
9.5.5 Hora do sistema e do hardware . . . 172
9.5.6 A infraestrutura de som . . . 173
9.5.7 desativar o protector de ecrã (screensaver) . . . 174
9.5.8 desativar os sons de beep . . . 175
9.5.9 Utilização da memória . . . 175
9.5.10 Segurança do sistema e verificação de integridade . . . 175
9.6 Dicas de armazenamento de dados . . . 176
9.6.1 Utilização do espaço em disco . . . 176
9.6.2 Configuração das partições do disco . . . 177
9.6.3 Aceder a partição a usar UUID . . . 177
9.6.4 LVM2 . . . 178
9.6.5 Configuração do sistema de ficheiros . . . 178
9.6.6 Criação do sistema de ficheiros e verificação de integridade . . . 179
9.6.7 Optimização do sistema de ficheiros por opções de montagem . . . 180
9.6.8 Optimização do sistema de ficheiros através do superblock . . . 180
9.6.9 Optimização do disco rígido . . . 181
9.6.10 Optimização de disco de estado sólido (SSD) . . . 181
9.6.11 Usar SMART para prever falhas no disco rígido . . . 181
9.6.12 Especifique o diretório de armazenamento temporário através de $TMPDIR . . . 181

9.6.13 Expandir o espaço de armazenamento utilizável via LVM . . . 182
9.6.14 Expandir o espaço de armazenamento utilizável ao montar outra partição . . . 182
9.6.15 Expandir o espaço de armazenamento utilizável ao fazer bind-mount para outro diretório . . . 182
9.6.16 Expansão do espaço de armazenamento utilizável ao fazer overlay-mounting para outro diretório 182
9.6.17 Expandir o espaço de armazenamento utilizável a usar ligações simbólicas . . . 182
9.7 A imagem de disco . . . 183
9.7.1 Criar o ficheiro de imagem de disco . . . 183
9.7.2 Escrever directamente no disco . . . 183
9.7.3 Montar o ficheiro de imagem de disco . . . 184
9.7.4 Limpar um ficheiro de imagem de disco . . . 185
9.7.5 Criar um ficheiro de imagem de disco vazio . . . 186
9.7.6 Criar o ficheiro de imagem ISO9660 . . . 186
9.7.7 Escrever directamente ao CD/DVD-R/RW . . . 187
9.7.8 Montar o ficheiro de imagem ISO9660 . . . 188
9.8 Os dados binários . . . 188
9.8.1 Ver e editar dados binários . . . 188
9.8.2 Manipular ficheiros sem montar o disco . . . 188
9.8.3 Redundância de dados . . . 188
9.8.4 Recuperação de ficheiros e dados e análise forense . . . 189
9.8.5 Dividir um ficheiro grande em ficheiros pequenos . . . 190
9.8.6 Limpar conteúdo de ficheiro . . . 190
9.8.7 Ficheiros dummy . . . 190
9.8.8 apagar um disco rígido inteiro . . . 190
9.8.9 Apagar uma área não utilizada do disco rígido . . . 191
9.8.10 Recuperar ficheiros apagados mas ainda abertos . . . 191
9.8.11 Procurar todas as ligações rígidas . . . 192
9.8.12 Consumo invisível do espaço do disco . . . 192
9.9 Dicas de encriptação de dados . . . 192
9.9.1 Encriptação de discos amovíveis com dm-crypt/LUKS . . . 193
9.9.2 Montar discos encriptados com dm-crypt/LUKS . . . 193
9.10 O kernel . . . 194
9.10.1 Parâmetros do kernel . . . 194
9.10.2 Cabeçalhos do kernel . . . 194
9.10.3 Compilar o kernel e módulos relacionados . . . 194
9.10.4 Compilar código-fonte do kernel: a recomendação da equipa do kernel de Debian . . . 195
9.10.5 Controladores de hardware e firmware . . . 196
9.11 Sistema virtualizado . . . 197
9.11.1 Ferramentas de virtualização e emulação . . . 197
9.11.2 Fluxo de trabalho da virtualização . . . 197
9.11.3 Montar o ficheiro de imagem de disco virtual . . . 199
9.11.4 Sistema chroot . . . 200
9.11.5 Sistemas de vários ambientes de trabalho . . . 201

10 Gestão de dados **202**

10.1 Partilhar, copiar e arquivar . . . 202
10.1.1 Ferramentas de arquivo e compressão . . . 203
10.1.2 Ferramentas de cópia de sincronização . . . 203
10.1.3 Idiomas para o arquivo . . . 203
10.1.4 Idiomas para a cópia . . . 205
10.1.5 Idiomas para a seleção de ficheiros . . . 206
10.1.6 Meio de arquivo . . . 207
10.1.7 Dispositivo de armazenamento amovível . . . 208
10.1.8 Escolha de sistema de ficheiros para partilhar dados . . . 209
10.1.9 Partilhar dados via a rede . . . 211
10.2 Salvaguarda (backup) e recuperação . . . 211
10.2.1 Política de cópia de segurança e recuperação . . . 211
10.2.2 Suites de utilitários de backup . . . 213
10.2.3 Sugestões de cópia de segurança . . . 214
10.2.3.1 Cópia de segurança GUI . . . 214
10.2.3.2 Cópia de segurança acionada por evento de montagem . . . 215
10.2.3.3 Cópia de segurança acionada por um evento de temporizador . . . 215
10.3 Infraestrutura da segurança de dados . . . 216
10.3.1 Gestão de chaves para GnuPG . . . 216
10.3.2 Usa GnuPG em ficheiros . . . 218
10.3.3 Usar GnuPG com o Mutt . . . 218
10.3.4 Usar GnuPG com o Vim . . . 218
10.3.5 O valor de controlo MD5 . . . 220
10.3.6 Gestor de palavras-passe . . . 220
10.4 Ferramentas de fusão de código fonte . . . 220
10.4.1 Extrair as diferenças para ficheiros fonte . . . 220
10.4.2 Fundir atualizações para ficheiros de fonte . . . 222
10.4.3 Integração interativa . . . 222
10.5 Git . . . 222
10.5.1 Configuração do cliente Git . . . 222
10.5.2 Comandos básicos do Git . . . 223
10.5.3 Dicas do Git . . . 224
10.5.4 Referências do Git . . . 224
10.5.5 Outros sistemas de controlo de versões . . . 226

11 Conversão de dados **227**
11.1 Ferramentas de conversão de dados em texto . . . 227
11.1.1 Converter um ficheiro de texto com o iconv . . . 228
11.1.2 Verifica ficheiro se é UTF-8 com o iconv . . . 229
11.1.3 Converter os nomes dos ficheiros com o iconv . . . 229
11.1.4 conversão EOL . . . 230
11.1.5 Conversão de TAB . . . 230
11.1.6 Editores com auto-conversão . . . 230
11.1.7 Extracção de texto simples . . . 231
11.1.8 Destacar e formatar dados de texto simples . . . 231
11.2 Dados XML . . . 231
11.2.1 Dicas básicas para XML . . . 233
11.2.2 Processamento de XML . . . 234
11.2.3 A extracção de dados de XML . . . 234
11.2.4 O lint de dados XML . . . 235
11.3 Formatação de texto . . . 235
11.3.1 formatação de texto roff . . . 235
11.3.2 TeX/LaTeX . . . 236
11.3.3 Impressão bonita de um manual . . . 237
11.3.4 Criar um manual . . . 237
11.4 Dados imprimíveis . . . 237
11.4.1 Ghostscript . . . 238
11.4.2 Juntar dois ficheiros PS ou PDF . . . 238
11.4.3 Utilitários de dados imprimíveis . . . 238
11.4.4 Imprimir com o CUPS . . . 238
11.5 A conversão de dados de mail . . . 240
11.5.1 Noções básicas de dados de mail . . . 240
11.6 Ferramentas de dados gráficos . . . 241
11.6.1 Ferramentas de dados gráficos (meta-pacote) . . . 241
11.6.2 Ferramentas gráficas de dados (GUI) . . . 241
11.6.3 Ferramentas de dados gráficos (CLI) . . . 243
11.7 Conversão de dados variados . . . 243

12 Programação **245**
12.1 O script de shell . . . 245
12.1.1 Compatibilidade da shell do POSIX . . . 246
12.1.2 Parâmetros da shell . . . 246
12.1.3 Condicionais da shell . . . 248
12.1.4 Ciclos (loops) da shell . . . 249

12.1.5 Variáveis de ambiente do shell . . . 249
12.1.6 A sequência de processamento da linha de comandos da shell . . . 249
12.1.7 Programas utilitários para script de shell . . . 250
12.2 Programação em linguagens interpretadas . . . 251
12.2.1 Depuração de códigos de linguagem interpretada . . . 252
12.2.2 Programa GUI com o script de shell . . . 252
12.2.3 Ações personalizadas para o arquivador GUI . . . 253
12.2.4 A loucura dos scripts curtos de Perl . . . 253
12.3 Codificação em linguagens compiladas . . . 253
12.3.1 C . . . 254
12.3.2 Programa C simples (gcc) . . . 254
12.3.3 Flex — um Lex melhor . . . 255
12.3.4 Bison — um Yacc melhor . . . 255
12.4 Ferramentas de análise de código estático . . . 257
12.5 Depuração . . . 257
12.5.1 Execução gdb básica . . . 257
12.5.2 Depurar o pacote Debian . . . 259
12.5.3 Obter um backtrace . . . 260
12.5.4 Comandos gdb avançados . . . 260
12.5.5 Verificar a dependência em bibliotecas . . . 261
12.5.6 Ferramentas dinâmicas de rastreio de chamadas . . . 261
12.5.7 Depurar Erros do X . . . 261
12.5.8 Ferramentas de detecção de fugas de memória . . . 261
12.5.9 Desassemblar binário . . . 261
12.6 Ferramentas de construção . . . 262
12.6.1 Make . . . 262
12.6.2 Autotools . . . 263
12.6.2.1 Compilar e instalar um programa . . . 263
12.6.2.2 Desinstalar um programa . . . 263
12.6.3 Meson . . . 264
12.7 Web . . . 264
12.8 A tradução do código-fonte . . . 264
12.9 Criar um pacote Debian . . . 265

A Apêndice **266**
A.1 o labirinto Debian . . . 266
A.2 História do Copyright . . . 266
A.3 Formato do documento . . . 267

Lista de Tabelas

1.1 Lista de pacotes de programas interessantes em modo de texto . . . 5
1.2 Lista de pacotes de documentação informativa . . . 5
1.3 Lista de utilização de diretórios chave . . . 8
1.4 Lista do primeiro caractere da saída de "ls -l" . . . 9
1.5 O modo numérico para permissões de ficheiros em comandos chmod(1) . . . 11
1.6 Exemplos do valor **umask** . . . 11
1.7 Lista de grupos notáveis disponibilizados pelo sistema para acesso a ficheiros . . . 13
1.8 Lista de grupos notáveis disponibilizados pelo sistema para execuções de comandos particulares . . . 13
1.9 Lista dos tipos de marcas temporais . . . 13
1.10 Lista de ficheiros de dispositivos especiais . . . 17
1.11 As teclas de atalho do MC . . . 19
1.12 A reacção à tecla enter no MC . . . 21
1.13 Lista de programas da shell . . . 22
1.14 Lista de teclas de atalho para bash . . . 23
1.15 Lista de operações do rato e ações-chave relacionadas no Debian . . . 24
1.16 Lista de teclas básicas do Vim . . . 25
1.17 lista dos comandos Unix básicos . . . 27
1.18 As 3 partes do valor locale . . . 29
1.19 Lista de recomendações de locale . . . 29
1.20 Lista de valores "$HOME" . . . 30
1.21 Padrões glob da shell . . . 31
1.22 Códigos de saída do comando . . . 32
1.23 Idiomas de comandos de shell . . . 33
1.24 Descritores de ficheiro predefinido . . . 34
1.25 Meta-caracteres para BRE e ERE . . . 36
1.26 A expressão de substituição . . . 37
1.27 Lista de trechos de script para canalizar comandos em pipe . . . 40

2.1 Lista de ferramentas de gestão de pacotes Debian . . . 42
2.2 Lista de sites de arquivos Debian . . . 45

2.3 Lista de área de arquivo Debian . . . 46
2.4 A relação entre suite e nome de código . . . 47
2.5 Lista de sites web chave para resolver problemas com um pacote específico . . . 51
2.6 Operações básicas de gestão de pacotes com a linha de comandos a utilizar apt(8), aptitude(8) e apt-get(8) /apt-cache(8) . . . 55
2.7 Opções de comando notáveis para o aptitude(8) . . . 55
2.8 Lista de teclas de atalho do aptitude . . . 56
2.9 Lista de vistas para o aptitude . . . 57
2.10 A categorização das vista de pacotes standard . . . 57
2.11 Lista da fórmula regex do aptitude . . . 59
2.12 Os ficheiros log para atividades de pacotes . . . 60
2.13 Lista de operações de gestão avançada de pacotes . . . 64
2.14 O conteúdo dos meta dados do arquivo Debian . . . 66
2.15 A estrutura de nomes dos pacotes Debian . . . 69
2.16 Os caracteres utilizáveis para cada componente nos nomes de pacotes Debian . . . 69
2.17 Ficheiros notáveis criados pelo `dpkg` . . . 70
2.18 Lista de valores notáveis de Pin-Priority para a técnica de **apt-pinning**. . . . 78
2.19 Lista de ferramentas proxy especiais para arquivos Debian . . . 82

3.1 Lista de gestores de arranque . . . 85
3.2 O significado da entrada de menu da parte acima de `/boot/grub/grub.cfg` . . . 86
3.3 Lista de utilitários de arranque para o sistema Debian . . . 88
3.4 Lista de níveis de erro do kernel . . . 92
3.5 Lista de trechos típicos do comando `journalctl` . . . 92
3.6 Lista de trechos de comandos típicos de gestão do `systemctl` . . . 94
3.7 Lista de outros trechos de comandos de monitorização `systemd` . . . 95

4.1 3 ficheiros de configuração importantes para pam_unix(8) . . . 98
4.2 A segunda entrada no conteúdo de "`/etc/passwd`" . . . 99
4.3 Lista de comandos para gerir informação de conta . . . 100
4.4 Lista de ferramentas para gerar palavras-passe . . . 101
4.5 Lista de sistemas PAM e NSS notáveis . . . 101
4.6 Lista de ficheiros de configuração acedidos pelo PAM e NSS . . . 102
4.7 Lista de serviços e portos inseguros e seguros . . . 104
4.8 Lista de ferramentas para disponibilizar medidas de segurança extra . . . 105

5.1 Lista de ferramentas de configuração de rede . . . 110
5.2 Lista de gamas de endereços de rede . . . 112
5.3 Tabela de tradução dos comandos obsoletos `net-tools` para os novos comandos `iproute2` . . . 115
5.4 Lista de comandos de rede de baixo nível . . . 115

5.5 Lista de ferramentas de optimização de rede . . . 116
5.6 Regras básicas para o valor MTU óptimo . . . 117
5.7 Lista de ferramentas de firewall . . . 118

6.1 Lista de exploradores web . . . 120
6.2 Lista de agentes utilizador de mail (MUA) . . . 122
6.3 Lista de pacotes básicos relacionados com o agente de transporte de correio . . . 124
6.4 Lista dos manuais importantes do postfix . . . 125
6.5 Lista de ficheiros de configuração relacionados com endereços de mail . . . 126
6.6 Lista de operações MTA básicas . . . 127
6.7 Lista de servidores de acesso remoto e utilitários . . . 128
6.8 Lista de ficheiros de configuração do SSH . . . 128
6.9 Lista de exemplos de arranque do cliente SSH . . . 129
6.10 Lista de clientes SSH para outras plataformas . . . 129
6.11 Lista de servidores de impressoras e utilitários . . . 131
6.12 Lista de outras aplicações de servidor de rede . . . 132
6.13 Lista de clientes de aplicação de rede . . . 133
6.14 Lista de RFCs populares . . . 133

7.1 Lista de ambientes de trabalho . . . 134
7.2 Lista de pacotes de infra-estruturas GUI notáveis . . . 136
7.3 Lista de aplicações GUI notáveis . . . 138
7.4 Lista de tipos de letra notáveis TrueType e OpenType . . . 139
7.5 Lista de fontes ambiente notáveis e pacotes relacionados . . . 140
7.6 Lista de ambientes sandbox notáveis e pacotes relacionados . . . 141
7.7 Lista de servidores de acesso remoto notáveis . . . 142
7.8 Lista de métodos de ligação ao servidor X . . . 142
7.9 Lista de programas relacionados com a manipulação da área de transferência de caracteres . . . 144

8.1 Lista do IBus e dos seus pacotes plugin . . . 150
8.2 Lista do Fcitx5 e dos seus pacotes plugin . . . 151

9.1 Lista de programas de apoio às atividades da consola . . . 154
9.2 Lista de ligações de teclas para o screen . . . 156
9.3 Informações sobre a inicialização do `vim` . . . 160
9.4 Lista de analisadores de log do sistema . . . 161
9.5 Mostrar exemplos de hora e data para o comando "`ls -l`" com o valor de **estilo de hora** . . . 162
9.6 Lista de ferramentas gráficas de manipulação de imagens . . . 163
9.7 Lista de pacotes que podem registar o histórico de configuração . . . 163
9.8 Lista de ferramentas para monitorizar e controlar as atividades de programas . . . 164

9.9 Lista de valores nice para a prioridade de agendamento . . . 165
9.10 Lista dos estilos do comando ps . . . 165
9.11 Lista dos sinais frequentemente usados para o comando kill . . . 169
9.12 Lista de teclas de comando SAK notáveis . . . 171
9.13 Lista de ferramenta de identificação de hardware . . . 172
9.14 Lista de ferramentas de configuração do hardware . . . 173
9.15 Lista de pacotes de som . . . 174
9.16 Lista de comandos para desativar o protector de ecrã . . . 174
9.17 Lista dos tamanhos de memória reportados . . . 175
9.18 Lista de ferramentas para segurança do sistema e verificação de integridade . . . 176
9.19 Lista de pacotes de gestão de partições do disco . . . 177
9.20 Lista de pacotes de gestão de sistemas de ficheiros . . . 179
9.21 Lista de pacote para ver e editar dados binários . . . 188
9.22 Lista de pacotes para ler e escrever ficheiros sem montar o disco . . . 189
9.23 Lista de ferramentas para adicionar redundância de dados a ficheiros . . . 189
9.24 Lista de pacotes para recuperação de ficheiros e dados e análise forense . . . 189
9.25 Lista de utilitários de encriptação de dados . . . 193
9.26 Lista de pacotes chave a serem instalados para a recompilação do kernel no sistema Debian . . . 195
9.27 Lista de ferramentas de virtualização . . . 198

10.1 Lista de ferramentas de arquivo e compressão . . . 204
10.2 Lista de ferramentas de cópia e sincronização . . . 205
10.3 Lista de hipóteses de sistemas de ficheiros para dispositivos de armazenamento amovíveis com cenários de utilização típica . . . 210
10.4 Lista de serviços de rede para escolher com o cenário de utilização típico . . . 211
10.5 Lista de suites utilitárias de salvaguarda . . . 213
10.6 Lista de ferramentas de infraestrutura da segurança de dados . . . 217
10.7 Lista de comandos do GNU Privacy Guard para gestão de chaves . . . 217
10.8 Lista do significado do código de confiança . . . 217
10.9 Lista de comandos do GNU Privacy Guard em ficheiros . . . 219
10.10 Lista de ferramentas de fusão de código fonte . . . 221
10.11 Lista de pacotes e comandos relacionados com o git . . . 223
10.12 Principais comandos do Git . . . 224
10.13 Dicas do Git . . . 225
10.14 Lista de outras ferramentas de sistemas de controlo de versões . . . 226

11.1 Lista de ferramentas de conversão de dados em texto . . . 227
11.2 Lista de valores de codificação e a utilização deles . . . 228
11.3 Lista de estilos EOL para diferentes plataformas . . . 230
11.4 Lista de comandos de conversão de TAB dos pacotes `bsdmainutils` e `coreutils` . . . 230

11.5 Lista de ferramentas para extracção de dados de texto simples . . . 232
11.6 Lista de ferramentas para destacar dados em texto simples . . . 232
11.7 Lista de entidades predefinidas para XML . . . 233
11.8 Lista de ferramentas XML . . . 234
11.9 Lista de ferramentas DSSSL . . . 234
11.10 Lista de ferramentas de extracção de dados de XML . . . 235
11.11 Lista de ferramentas de impressão bonita de XML . . . 235
11.12 Lista de ferramentas de formatação de texto . . . 236
11.13 Lista de pacotes para ajudar a criar o manual (manpage) . . . 237
11.14 Lista de interpretadores PostScript Ghostscript . . . 238
11.15 Lista de utilitários de dados imprimíveis . . . 239
11.16 Lista de pacotes para ajudar na conversão de dados de mail . . . 240
11.17 Lista de ferramentas de dados gráficos (meta-pacote) . . . 241
11.18 Lista de ferramentas de dados gráficos (GUI) . . . 242
11.19 Lista de ferramentas de dados gráficos (CLI) . . . 244
11.20 Lista de ferramentas de conversão de dados variados . . . 244

12.1 Lista dos 'bashisms' típicos . . . 246
12.2 Lista de parâmetros da shell . . . 247
12.3 Lista de expansões de parâmetros de shell . . . 247
12.4 Lista de substituições de parâmetros de shell chave . . . 247
12.5 Lista de operadores de comparação de ficheiros na expressão condicional . . . 248
12.6 Lista de operadores de comparação de cadeias na expressão condicional . . . 249
12.7 Lista de pacotes que contém programas utilitários pequenos para scripts de shell . . . 251
12.8 Lista de pacotes relacionados com o interpretador . . . 251
12.9 Lista de programas de diálogo . . . 252
12.10 Lista de pacotes relacionados com o compilador . . . 254
12.11 Lista de geradores de análise LALR compatíveis com Yacc . . . 255
12.12 Lista de ferramentas para análise de código estático . . . 258
12.13 Lista de pacotes de depuração . . . 258
12.14 Lista de comandos gdb avançados . . . 260
12.15 Lista de ferramentas de detecção de fugas de memória . . . 261
12.16 Lista de pacotes de ferramentas de compilação . . . 262
12.17 Lista de variáveis automáticas do make . . . 262
12.18 Lista de expansões da variável do make . . . 263
12.19 Lista de ferramentas de tradução de código-fonte . . . 265

Resumo

Este livro é livre; pode redistribuí-lo e/ou modificá-lo sob os termos da Licença Pública Geral GNU de qualquer versão compatível com a Definição Debian de Software Livre (DFSG).

Prefácio

Esta Referência Debian (version 2.152) (2026-10-10 14:56:25 UTC) destina-se a fornecer uma visão geral da administração do sistema Debian como um guia do utilizador pós-instalação.

O leitor alvo é quem está disposto a aprender scripts shell, mas que não está pronto para ler todas as fontes C para descobrir como o sistema GNU/Linux funciona.

Para instruções de instalação, veja:

- Guia de Instalação de Debian GNU/Linux para o sistema atualmente stable
- Guia de Instalação de Debian GNU/Linux para o sistema atualmente stable

Aviso Legal

Todas as garantias são recusadas. Todas as marcas registadas são propriedade dos respetivos proprietários de marcas registadas deles.

O próprio sistema Debian é um alvo em movimento. Isso torna sua documentação difícil de estar atualizada e correta. Embora a versão atual de `testing` do sistema Debian tenha sido usada como base para escrever isto, alguns conteúdos podem já estar desatualizados no momento que você ler isto.

Por favor, trate este documento como a referência secundária. Este documento não substitui nenhum guia autorizado. O autor e os colaboradores não se responsabilizam por consequências de erros, omissões ou ambiguidade neste documento.

O que é Debian

O Projeto Debian é uma associação de indivíduos que fizeram causa comum para criar um sistema operacional livre. A distribuição dele é caracterizada pelo seguinte.

- Compromisso com a liberdade do software: Contrato Social Debian e Definição Debian de Software Livre (DFSG)
- Esforço distribuído de voluntários não remunerados através da Internet: https://www.debian.org
- Grande quantidade de pacotes de software pré-compilados de alta qualidade
- Foco em estabilidade e segurança com acesso fácil a atualizações de segurança
- Centra-se na atualização suave para os pacotes de software mais recentes nos arquivos `testing`
- Grande quantidade de arquitecturas de hardware suportadas

As peças de Software Livre em Debian vêm de GNU, Linux, BSD, X, ISC, Apache, Ghostscript, Common Unix Printing System , Samba, GNOME, KDE, Mozilla, LibreOffice, Vim, TeX, LaTeX, DocBook, Perl, Python, Tcl, Java, Ruby, PHP, Berkeley DB, MariaDB, PostgreSQL, SQLite, Exim, Postfix, Mutt, FreeBSD, OpenBSD, Plan 9 e muitos mais projectos de software livre independentes. Debian integra esta diversidade de Software Livre num sistema.

Acerca deste documento

Regras orientadoras

Foram seguidas as seguintes regras de orientação ao compilar este documento.

- Dar uma visão geral e saltar casos não comuns. (**Imagem Geral**)
- Manter Curto e Simples. (Principio **KISS**)
- Não reinventar a roda. (Utilizar apontadores para **as referências existentes**)
- Foco nas ferramentas não-GUI e consolas. (Utilizar **exemplos de shell**)
- Ser objetivo. (Utilizar popcon etc.)

Dica
Tentei elucidar aspectos hierárquicos e níveis mais baixos do sistema.

Pré-requisitos



Atenção
Espera-se que se esforce a procurar respostas por si próprio e para além desta documentação. Este documento apenas oferece pontos de arranque eficientes.

Tem de procurar a solução por si a partir de fontes primárias.

- O site Debian em https://www.debian.org para informação geral
- A documentação sob o diretório "`/usr/share/doc/`*`nome_do_pacote`*"
- O **manual** de estilo Unix: "`dpkg -L` *`package_name`* `|grep '/man/man.*/'`"
- A **página info** estilo GNU: "`dpkg -L` *`package_name`* `|grep '/info/'`"
- O relatório de erros: https://bugs.debian.org/*package_name*
- O Debian Wiki em https://wiki.debian.org/ para os tópicos específicos e em movimento
- A Especificação Única do UNIX de Grupos Abertos Página do Sistema UNIX
- A enciclopédia livre Wikipedia em https://www.wikipedia.org/
- O Livro de Mão dos Administradores de Debian
- Os HOWTOs de O Projeto de Documentação do Linux (TLDP)

Nota
Para documentação detalhada, pode necessitar instalar o correspondente pacote de documentação chamado com o sufixo "`-doc`".

Convenções

Este documento fornece informação através do seguinte estilo de apresentação simplificado com exemplos de comandos de shell bash(1).

```
# command-in-root-account
$ command-in-user-account
```

Estas 'prompts' da shell distinguem a conta utilizada e correspondem a definir variáveis de ambiente como: "`PS1='\$'`" e "`PS2=' '`". Estes valores são escolhidos para bem da legibilidade deste documento e não são típicos do sistema instalado.

Todos os exemplos de comandos são executados no formato regional Inglês "`LANG=en_US.UTF8`". Não espere que as cadeias de caracteres de espaço reservado, tais como *`command-in-root-account`* e *`command-in-user-account`* sejam traduzidas em exemplos de comandos. Esta é uma escolha intencional para manter todos os exemplos traduzidos actualizados.

Nota
Veja o significado das variáveis de ambiente "$PS1" e "$PS2" em bash(1).

A **ação** necessária do administrador do sistema é escrita em sentido imperativo, p.e. "Carregue na tecla Enter após escrever cada cadeia de comando na shell."

A coluna **descrição** e semelhantes na tabela podem conter um **sintagma nominal** seguido da convenção de descrição curta do pacote que deixa cair os artigos como "um" e "o". Pode em alternativa conter uma frase no infinitivo tal como um **sintagma nominal** sem o antecedente "para" a seguir a convenção de descrição curta de comando das 'manpages'. Isto pode parecer esquisito para algumas pessoas mas são as minhas escolhas intencionais de estilo para manter esta documentação o mais simples possível. Estes **sintagmas nominais** não começam por maiúscula nem terminam com ponto final a seguir esta convenção de descrição curta.

Nota
Substantivos próprios incluindo os nomes de comandos mantêm maiúscula/minúscula sem respeitarem a sua localização.

Um **bloco de comandos** citado num parágrafo de texto é referido pelo tipo de letra de dactilografia entre aspas, tal como "`aptitude safe-upgrade`".

Os **dados em texto** de um ficheiro de configuração citados num parágrafo de texto são referidos em tipo de letra de máquina de escrever entre aspas, tal como "`deb-src`".

Um **comando** é referenciado pelo seu nome em tipo de letra de máquina de escrever seguido opcionalmente pelo número de secção da manpage em parêntesis, tal como bash(1). É encorajado a obter informação ao escrever o seguinte.

```
$ man 1 bash
```

Uma **manpage** é referida pelo seu nome em tipo de letra de máquina de escrever seguido pelo número de secção dele da manpage em parêntesis, tal como sources.list(5). É encorajado a obter informação ao escrever o seguinte.

```
$ man 5 sources.list
```

Uma **página info** é referenciada pelo seu comando em tipo de letra de máquina de escrever entre aspas, tal como "`info make`". É encorajado a obter informação ao escrever o seguinte.

```
$ info make
```

Um **nome de ficheiro** é referenciado em tipo de letra de máquina de escrever entre aspas, tal como "`/etc/passwd`". Para os ficheiros de configuração, é encorajado a obter informação ao escrever o seguinte.

```
$ sensible-pager "/etc/passwd"
```

Um **nome de diretório** é referenciado em tipo de letra de máquina de escrever entre aspas, tal como "`/etc/apt/`". É encorajado a explorar o conteúdo dele ao escrever o seguinte.

```
$ mc "/etc/apt/"
```

Um **nome de pacote** é referenciado pelo nome dele em tipo de letra de máquina de escrever, tal como `vim`. É encorajado a obter informação ao escrever o seguinte.

```
$ dpkg -L vim
$ apt-cache show vim
$ aptitude show vim
```

Uma **documentação** pode indicar a localização dela pelo nome de ficheiro em tipo de letra de dactilografia entre aspas, tal como "`/usr/share/doc/base-passwd/users-and-groups.txt.gz`" e "`/usr/share/doc/base-passwd/`" ou pelo seu URL, tal como https://www.debian.org. É encorajado a ler a documentação ao escrever o seguinte.

```
$ zcat "/usr/share/doc/base-passwd/users-and-groups.txt.gz" | sensible-pager
$ sensible-browser "/usr/share/doc/base-passwd/users-and-groups.html"
$ sensible-browser "https://www.debian.org"
```

Uma **variável de ambiente** é referenciada pelo seu nome com um "$" inicial em tipo de letra de máquina de escrever, entre aspas, tal como "`$TERM`". É encorajado a obter o valor atual dele ao escrever o seguinte.

```
$ echo "$TERM"
```

popcon

Os dados popcon são apresentados como a medida objectiva da popularidade de cada pacote. Foi descarregado em 2026-10-07 05:10:21 UTC e contém a submissão total de 291550 relatórios sobre 224298 pacotes binários e 25 arquitecturas.

Nota
Por favor note que o arquivo `amd64 unstable` contém atualmente apenas 78072 pacotes. Os dados popcon contém relatórios de muitas instalações de sistemas antigos.

O número de popcon precedido de "V:" para "votos" é calculado por "1000 * (as submissões popcon para o pacote executado recentemente no PC) / (o total de submissões de popcon)".

O número de popcon precedido de "I:" para "instalações" é calculado por "1000 * (as submissões popcon para o pacote instalado no PC) / (o total de submissões de popcon)".

Nota
As figuras do popcon não devem ser consideradas como medidas absolutas da importância dos pacotes. Existem muitos factores que podem desviar as estatísticas. Por exemplo, um sistema que participa no popcon pode ter diretórios montados como o "`/usr/bin`" com a opção "`noatime`" para melhoria da performance do sistema e efectivamente desativar os "votos" de tal sistema.

O tamanho do pacote

Os dados de tamanho do pacote são também apresentados como a medida objectiva para cada pacote. São baseados no "`Installed-Size:`" reportado pelo comando "`apt-cache show`" ou pelo "`aptitude show`" (atualmente na `amd64` arquitectura para o lançamento `unstable`). O tamanho reportado está em KB (Kilobyte = unidade para 1024 bytes).

Nota
Um pacote com um tamanho de pacote numericamente pequeno pode indicar que o pacote no lançamento `unstable` é um pacote dummy que instala outros pacotes com conteúdos significativos por dependência. O pacote dummy activa uma transição suave ou divisão do pacote.

Nota
Um tamanho de pacote seguido por "(*)" indica que o pacote no lançamento `unstable` está em falta e em vez dele é usado o tamanho do pacote para o lançamento `experimental`.

Relatórios de bugs deste documento

Se encontrar quaisquer problemas neste documento por favor preencha um relatório de bug contra o pacote `debian-refer` a utilizar o reportbug(1) . Por favor inclua sugestões de correção com "`diff -u`" sobre a versão de texto ou código-fonte.

Memorandos para novos utilizadores

Aqui estão alguns memorandos para os novos utilizadores:

- Faça salvaguardas dos seus dados
  - Veja Secção 10.2.
- Mantenha a sua palavra-passe e chaves de segurança seguras
- KISS (keep it simple stupid- manté-lo simples estupido)
  - Não exageres na engenharia do teu sistema
- Leia os seus ficheiros log
  - O **PRIMEIRO** erro é aquele que conta
- RTFM (read the fine manual - leia o manual)
- Pesquise na Internet antes de fazer perguntas
- Não seja root quando não precisa de o ser
- Não brinque com o sistema de gestão de pacotes
- Não escreve nada que não compreenda
- Não altere as permissões do ficheiro (antes da revisão de segurança completa)
- Não abandones a shell de root antes de **TESTARES** as tuas alterações
- Tenha sempre uma média de arranque alternativo (Memória flash USB, CD-ROM, …)

Algumas citações para os novos utilizadores

Aqui estão algumas citações interessantes da lista de email Debian que podem ajudar a elucidar novos utilizadores.

- ”Isto é Unix. Dá-lhe corda suficiente para você se enforcar.” --- Miquel van Smoorenburg <miquels at cistron.nl>
- ”Unix É amigo do utilizador... Apenas é selectivo sobre quem são os seus amigos.” --- Tollef Fog Heen <tollef at add.no>

A Wikipedia tem o artigo ”Filosofia do Unix” que lista citações interessantes.

Capítulo 1

Manuais de GNU/Linux

Acho que aprender um sistema de computador é como aprender uma nova língua estrangeira. Apesar dos livros e documentação darem ajuda, tem que praticar também. De modo a ajudá-lo a iniciar suavemente, elaborei alguns pontos básicos.

O design poderoso da Debian GNU/Linux vem do sistema operativo Unix, isto é, um sistema operativo multi-utilizador e multi-tarefa. Necessita aprender a tirar vantagem do poder destas funcionalidades e semelhanças entre Unix e GNU/Linux.

Não se esconda dos textos orientados ao Unix e não se guie somente nos textos de GNU/Linux, por isto rouba-lhe muita informação útil.

Nota
Se tem usado qualquer sistema de tipo Unix com ferramentas de linha de comandos, provavelmente já sabe tudo o que explico aqui. Por favor use isto como um teste de realidade e refrescamento.

1.1 Básico da consola

1.1.1 A linha de comandos da shell

Ao iniciar o sistema, é-lhe apresentado o ecrã de autenticação baseado em caracteres se não instalou nenhum ambiente GUI como o GNOME ou o sistema de desktop KDE. Suponha que o seu nome de anfitrião é `foo`, a prompt de login tem o seguinte aspeto.

Se instalou um ambiente GUI, então pode ir à mesma para uma prompt de login baseada em caracteres ao pressionar Ctrl-Alt-F3 e pode regressar ao ambiente GUI via Ctrl-Alt-F2 (para mais informação veja Secção 1.1.6 em baixo).

```
foo login:
```

Na prompt de login, escreva o seu nome de utilizador, p.e. `penguin` e carregue na tecla Enter, depois escreva a sua palavra-passe e carregue novamente na tecla Enter.

Nota
A seguir a tradição do Unix, o nome de utilizador e palavra-passe do sistema Debian são sensíveis a maiúsculas/minúsculas. O nome de utilizador é geralmente escolhido apenas em minúsculas. A primeira conta de utilizador é normalmente criada durante a instalação. Podem ser criadas contas de utilizador adicionais com adduser(8) pelo root.

O sistema inicia com a mensagem de boas vindas armazenada em "`/etc/motd`" (Mensagem do Dia) e apresenta uma prompt de comandos.

```
Debian GNU/Linux 12 foo tty3

foo login: penguin
Password:

Linux foo 6.5.0-0.deb12.4-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.5.10-1~bpo12+1 (2023-11-23) ↩
     x86_64

The programs included with the Debian GNU/Linux system are free software;
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.

Last login: Wed Dec 20 09:39:00 JST 2023 on tty3
foo:~$
```

Está agora na shell. A shell interpreta os seus comandos.

1.1.2 A linha de comandos na GUI

Se instalou um ambiente GUI durante a instalação, é-lhe apresentado o ecrã gráfico de início de sessão ao iniciar o sistema. Introduza o seu nome de utilizador e a sua palavra-passe para iniciar sessão na conta de utilizador não privilegiado. Utilize o separador para navegar entre o nome de utilizador e a palavra-passe, ou utilize o clique primário do rato.

Pode obter a linha de comandos da shell num ambiente GUI iniciando um programa `x-terminal-emulator` como o gnome-terminal(1), rxvt(1) ou xterm(1). No ambiente de trabalho GNOME, premir a tecla SUPER (tecla Windows) e escrever "terminal" na linha de pesquisa faz o truque.

Em alguns Ambientes de Trabalho (como o `fluxbox`), pode não existir um ponto de partida óbvio para o menu. Se isto acontecer, tente clicar (botão direito) no fundo do ambiente de trabalho e espere que apareça um menu.

1.1.3 A conta root

A conta root também é chamada como de super utilizador ou de utilizador privilegiado. A partir desta conta, pode executar as seguintes tarefas de administração do sistema:

- Ler, escrever e remover quaisquer ficheiros no sistema independentemente das permissões deles
- Definir o dono e permissões de quaisquer ficheiros no sistema
- Definir a palavra-passe de quaisquer utilizadores não privilegiados do sistema
- Iniciar sessão (Login) em qualquer conta sem a palavra-passe deles

Este poder ilimitado da conta root requer que você seja atento e responsável quando a utilizar.



Atenção
Nunca partilhe a palavra-passe de root com outros.

Nota
As permissões de um ficheiro (incluindo dispositivos de hardware como CD-ROM etc. os quais são apenas outros ficheiros para o sistema Debian) podem torná-lo não-utilizável ou inacessível para utilizadores não-root. Apesar da utilização da conta root ser um modo rápido de testar este tipo de situação, a resolução dela deve ser feita através da definição correcta das permissões do ficheiro e membros dos grupos de utilizadores. (veja Secção 1.2.3).

1.1.4 A linha de comandos shell do root

Aqui estão alguns métodos básicos de obter o prompt da shell de root ao utilizar a palavra-passe do root:

- Escreva `root` na prompt de login baseada em caracteres.
- Escreva "`su -l`" na prompt da shell de qualquer utilizador.
  - Isto não preserva o ambiente do utilizador atual.
- Escreva "`su`" na prompt de shell de qualquer utilizador.
  - Isto preserva algum do ambiente do utilizador atual.

1.1.5 GUI de ferramentas de administração do sistema

Quando o menu do ambiente de trabalho não inicia automaticamente as ferramentas de administração do sistema GUI com o privilégio apropriado, pode iniciá-las a partir da prompt da shell de raiz do emulador de terminal, como o gnome-terminal(1), rxvt(1) ou xterm(1). Ver Secção 1.1.4 e Secção 7.9.



Atenção
Nunca inicie o gestor de ecrã/sessão GUI com a conta root ao escrever `root` na prompt do gestor de ecrã/sessão como o gdm3(1).
Nunca execute programas GUI remotos que não sejam confiáveis no X Window quando é mostrada informação critica porque pode "espiar" o seu ecrã X.

1.1.6 Consolas virtuais

Por predefinição no sistema Debian existem disponíveis seis consolas de caracteres alternáveis estilo-VT100 para arrancar a shell de comandos diretamente na máquina Linux. A menos que esteja num ambiente GUI, pode mudar entre consolas virtuais ao pressionar a `Tecla-Alt-esquerda` e simultaneamente numa das teclas `F1` — `F6`. Cada consola de caracteres permite um login independente à conta e oferece um ambiente multi-utilizador. Este ambiente multi-utilizador é uma funcionalidade excelente do Unix e muito viciante.

Se está no ambiente GUI, pode ganhar acesso à consola 3 ao pressionar as teclas `Ctrl-Alt-F3`, isto é, a `left-Ctrl-key` a `left-Alt-key` e a tecla `F3-key` pressionadas em conjunto. Pode regressar ao ambiente GUI, que normalmente executa na consola virtual 2, ao pressionar `Alt-F2`.

Pode, em alternativa, mudar para outra consola virtual, por exemplo à consola 3, a partir da linha de comandos.

```
# chvt 3
```

1.1.7 Como abandonar a linha de comandos

Escreva `Ctrl-D`, isto é, a `left-Ctrl-key` e a `d-key` pressionadas ao mesmo tempo, na linha de comandos para fechar a atividade da shell. Se estiver na consola de caracteres, com isto, retorna ao aviso de login. Mesmo que estes caracteres de controle sejam referidos como ”control D” com letra maiúscula, não precisa de pressionar a tecla Shift. A expressão curta, `^D`, também é usada para `Ctrl-D`. Em alternativa pode escrever ”exit”.

Se estiver no x-terminal-emulator(1), com isto pode fechar a janela do `x-terminal-emulator`.

1.1.8 Como desligar o sistema

Tal como qualquer outro SO moderno onde operar ficheiros involve pôr dados em cache em memória para melhorar a performance, o sistema Debian precisa de um processo apropriado de desligar antes que a energia possa ser, em segurança, desligada. Isto é para manter a integridade dos ficheiros, ao forçar todas as alterações em memória a serem escritas no disco. Se estiver disponível software de controle de energia, o processo de desligar desliga automaticamente a energia do sistema. (Caso contrário, pode ter de pressionar o botão de energia por alguns segundos após o procedimento de desligar.)

Pode desligar o sistema sob o modo normal de multi-utilizador a partir da linha de comandos.

```
# shutdown -h now
```

Pode desligar o sistema sob o modo único-utilizador a partir da linha de comandos.

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

Veja a Secção 6.3.8.

1.1.9 Recuperar uma consola sã

Quando o ecrã fica estranho após fazer coisas estranhas tal como ”`cat` *`some-binary-file`*”, escreva ”`reset`” na linha de comandos. Poderá não ver o comando a aparecer quando o escreve. Também pode utilizar ”`clear`” para limpar o ecrã.

1.1.10 Sugestões de pacotes adicionais para o novato

Apesar de mesmo uma instalação mínima do sistema Debian sem quaisquer tarefas de ambiente de trabalho disponibilizar as funcionalidades básicas do Unix, é uma boa ideia instalar alguns pacotes baseados em linha de comandos e terminais de caracteres baseados em curses tais como o `mc` e o `vim` com o apt-get(8) para os iniciantes começarem, pelo seguinte.

```
# apt-get update
 ...
# apt-get install mc vim sudo aptitude
 ...
```

Se já tiver estes pacotes instalados, não serão instalados novos pacotes.

Pode ser uma boa ideia ler algumas documentações informativas.

Pode instalar alguns destes pacotes com o seguinte.

```
# apt-get install package_name
```

pacote	popcon	tamanho	descrição
mc	V:40, I:176	1590	Um gestor de ficheiro de ecrã completo em modo de texto
sudo	V:761, I:869	6814	Um programa para permitir privilégios de root limitados aos utilizadores
vim	V:83, I:338	4164	O editor de texto de Unix Vi IMproved, um editor de texto para programadores (versão standard)
vim-tiny	V:59, I:980	1867	O editor de texto de Unix Vi IMproved, um editor de texto para programadores (versão compacta)
emacs-nox	V:3, I:12	46564	Emacs do Projecto GNU, o editor de texto extensível baseado em Lisp
w3m	V:10, I:129	2853	Navegadores de WWW de modo de texto
gpm	V:8.3, I:9.0	524	O cortar-e-colar estilo Unix na consola de texto (daemon)

Tabela 1.1: Lista de pacotes de programas interessantes em modo de texto

pacote	popcon	tamanho	descrição
doc-debian	I:886	185	Documentação do Projecto Debian, (FAQ do Debian) e outros documentos
debian-policy	I:6.8	5147	Manual de Políticas Debian e documentos relacionados
developers-reference	V:0.2, I:2.2	2657	Guias e informação para programadores de Debian
debmake-doc	I:0.33	11803	Guia para Mantedor Debian
debian-history	I:0.50	6513	História do Projecto Debian
debian-faq	I:885	791	FAQ do Debian

Tabela 1.2: Lista de pacotes de documentação informativa

1.1.11 Uma conta de utilizador extra

Se não deseja usar a sua conta de utilizador principal para as seguintes atividades de treino, pode criar uma conta de utilizador para treinos, por exemplo, `fish`, a fazer o seguinte:

```
# adduser fish
```

Responder a todas as questões.

Isto cria uma conta chamada `fish`. Após praticar, pode remover esta conta de utilizador e o diretório home dele a fazer o seguinte:

```
# deluser --remove-home fish
```

Em sistemas não-Debian e Debian especializados, as actividades acima precisam de utilizar utilitários de baixo nível useradd(8) e userdel(8).

1.1.12 Configuração do sudo

Para a típica estação de trabalho de um único utilizador como o ambiente de trabalho do sistema Debian no PC portátil, é comum implementar uma configuração simples do sudo(8) como a seguir para permitir ao utilizador não-privilegiado, ex. `penguin`, ganhar privilégios administrativos apenas com a sua palavra-passe de utilizador mas sem a palavra-passe do root.

```
# echo "penguin  ALL=(ALL) ALL" >> /etc/sudoers
```

Em alternativa, é também comum fazer como a seguir para permitir a um utilizador não privilegiado, ex. `penguin`, ganhar privilégios administrativos sem qualquer palavra-passe.

```
# echo "penguin  ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers
```

Este truque só deve ser usado na estação de trabalho de um único utilizador que administra e onde é o único utilizador.



Atenção
Não configure assim as contas de utilizadores normais numa estação de trabalho multi-utilizador porque seria muito mau para a segurança do sistema.



Cuidado
A palavra-passe e a conta `penguin` no exemplo em cima requer tanta protecção como a palavra-passe do root e a conta do root.
O privilégio administrativo neste contexto pertence a alguém autorizado a executar as tarefas de administração do sistema numa estação de trabalho. Nunca dê tais privilégios a um gestor do departamento Administrativo da sua firma ou ao seu chefe a menos que eles sejam autorizados e capazes.

Nota
Para disponibilizar privilégios de acesso a dispositivos limitados e ficheiros limitados, deve considerar usar o **group** para disponibilizar acesso limitado em vez de usar os privilégios do `root` via sudo(8).
Com uma configuração melhor pensada e cuidada, o sudo(8) pode garantir privilégios administrativos limitados a outros utilizadores num sistema partilhado sem partilhar a palavra-passe do root. Isto pode ajudar com as responsabilidades com máquinas com múltiplos administradores para que possa saber quem fez o quê. Por outro lado, pode querer que mais ninguém tenha tais privilégios.

1.1.13 Hora de brincar

Agora está pronto para brincar com o sistema Debian sem riscos desde que use a conta de utilizador sem-privilégios.

Isto porque o sistema Debian é, mesmo após uma instalação predefinida, configurado com permissões de ficheiros apropriadas que previne os utilizadores não privilegiados de danificarem o sistema. É claro, podem ainda existir alguns buracos que possam ser explorados mas aqueles que se preocupam com estes problemas não deveriam ler esta secção e deveriam ler o Manual de Segurança Debian.

Aprendemos o sistema Debian como um sistema tipo Unix com o seguinte:

- Secção 1.2 (conceitos básicos)
- Secção 1.3 (método de sobrevivência)
- Secção 1.4 (método básico)
- Secção 1.5 (mecanismo da shell)
- Secção 1.6 (método de processamento de texto)

1.2 Sistema de ficheiros tipo Unix

No GNU/Linux e noutros sistemas operativos tipo Unix, os ficheiros estão organizados em diretórios. Todos os ficheiros e diretórios estão organizados numa grande árvore que nasce em "`/`". É chamada uma árvore porque se desenhar o sistema de ficheiros, parece-se com uma árvore mas está de cabeça para baixo.

Estes ficheiros e diretórios podem estar espalhados por vários dispositivos. mount(8) serve para anexar o sistema de ficheiros encontrado num dispositivo à grande árvore de ficheiros. Reciprocamente, umount(8) desanexa-os novamente. Nos kernel Linux recentes, o mount(8) com algumas opções pode unir parte de uma árvore de ficheiros noutro lugar ou pode montar um sistema de ficheiros como partilhado, privado, escravo ou não-unível. As opções do mount suportadas para cada sistema de ficheiros estão disponíveis em "`/usr/share/doc/linux-doc-*/Documentation/fi`

Os **diretórios** no sistema Unix são chamados **pastas** nalguns outros sistemas. Por favor note também que não existe conceito para **drive** tal como "`A:`" em qualquer sistema Unix. Existe um sistema de ficheiros e tudo está incluído nele. Esta é uma enorme vantagem em comparação com o Windows.

1.2.1 Noções básicas de ficheiros Unix

Aqui estão algumas noções básicas de ficheiros Unix:

- Os nomes de ficheiro são **sensíveis a maiúsculas/minúsculas**. Isto é, "`MYFILE`" e "`MyFile`" são ficheiros diferentes.
- O **diretório raiz** significa a raiz do sistema de ficheiros e é referido simplesmente como "`/`". Não confundir isto com o diretório pessoal do utilizador root: "`/root`".
- Todos os diretórios têm um nome que pode conter quaisquer letras ou símbolos **excepto "/"**. O diretório raiz é uma excepção. O nome dele é "`/`" (pronuncia-se "slash" ou "o diretório raiz") e não pode ser renomeado.
- Cada ficheiro ou diretório é designado por um **nome de ficheiro totalmente qualificado**, **nome de ficheiro absoluto**, ou **caminho**, que fornece a sequência de diretórios que têm de ser percorridos para o alcançar. Estes três termos são sinónimos.
- Todos os **nomes de ficheiro totalmente qualificados** começam com o diretório "`/`" e existe um "`/`" entre cada diretório ou ficheiro no nome do ficheiro. O primeiro "`/`" é o diretório de nível de topo e os outros "`/`"' separam sucessivamente os sub-diretórios, até que se chegue à última entrada que é o nome real do ficheiro. As palavras utilizadas aqui conseguem ser confusas. Veja o seguinte **nome de ficheiro completamente qualificado** como um exemplo: "`/usr/share/keytables/us.map.gz`". No entanto, as pessoas também se referem ao seu nome base sozinho "`us.map.gz`" como um nome de ficheiro.
- O diretório raiz tem algumas ramificações, tais como "`/etc/`" e "`/usr/`". Estes sub-diretórios por sua vez ramificam-se em mais sub-diretórios, tais como "`/etc/systemd/`" e "`/usr/local/`". O todo, visto em conjunto, é a chamada **árvore de diretórios**. Pode pensar num nome de ficheiro absoluto como um caminho desde a base da árvore ("`/`") até ao fim de um ramo (um ficheiro). Também pode ouvir pessoas falar da árvore de diretórios como se fosse uma árvore de **família** a juntar todos os descendentes diretos numa única figura chamada de diretório raiz ("`/`"): assim, os sub-diretórios têm **pais** e um caminho mostra a linhagem completa de um ficheiro. Existem também caminhos relativos que começam algures noutro ponto que não o diretório raiz. Deve lembrar-se que o diretório "`../`" refere-se ao diretório pai. Esta terminologia também se aplica a outras estruturas semelhantes a diretórios, como estruturas hierárquicas de dados.
- Não existe componente especial no nome de caminho que corresponde a um dispositivos físico, tal como o seu disco rígido. Isto difere de RT-11, CP/M, OpenVMS, MS-DOS, AmigaOS e Microsoft Windows, onde o caminho contém um nome de dispositivo tal como "`C:\`". (No entanto, existem entradas nos diretórios que referem-se a dispositivos físicos como parte do sistema de ficheiros normal. Veja Secção 1.2.2.)

Nota
Apesar de **poder** usar quase todas as letras ou símbolos num nome de ficheiro, na prática é má ideia fazê-lo. É melhor evitar quaisquer caracteres que geralmente têm significados especiais na linha de comandos, incluindo espaços, tabs, novas linhas e outros caracteres especiais: `{ } ( ) [ ] ' ` " \ / > < | ; ! # & ^ * % @ $` . Se deseja separar palavras num nome, as boas escolhas são o ponto, traço e underscore. Também pode capitalizar cada palavra assim "`LikeThis`". Os utilizadores avançados de Linux procuram evitar espaços nos nomes de ficheiros.

Nota
A palavra ”root” pode significar o ”utilizador root” ou o ”diretório raiz (root)”. O contexto da utilização deles deve torná-lo claro.

Nota
A palavra **caminho (path)** é usada não apenas para o **nome-de-ficheiro totalmente qualificado** como em cima mas também para o **caminho de busca de comandos**. O significado pretendido é geralmente claro a partir do contexto.

As melhores práticas detalhadas para a hierarquia de ficheiros estão descritas no Filesystem Hierarchy Standard (”`/usr/share/doc/debian-policy/fhs/fhs-2.3.txt.gz`” e hier(7)). Deve lembrar-se dos seguintes factos como princípio.

diretório	**utilização do diretório**
`/`	o diretório raiz
`/etc/`	ficheiros de configuração de todo o sistema
`/var/log/`	ficheiros log do sistema
`/home/`	todos os diretórios home de todos os utilizadores não privilegiados

Tabela 1.3: Lista de utilização de diretórios chave

1.2.2 Internos do sistema de ficheiros

A seguir a **tradição do Unix**, o sistema Debian GNU/Linux disponibiliza o sistema de ficheiros sob o qual residem os dados físicos em discos rígidos e outros dispositivos de armazenamento e a interacção com os dispositivos de hardware como ecrãs de consola e consolas série remotas são representados num modo unificado sob ”`/dev/`”.

Cada ficheiro, diretório, ’named pipe’ (um modo de dois programas partilharem dados), ou dispositivo físico num sistema Debian GNU/Linux tem uma estrutura de dados chamada inode que descreve os seus atributos associados como o utilizador que o possui (o dono), o grupo a que pertence, a hora do último acesso, etc. A ideia de representar praticamente tudo no sistema de ficheiros foi uma inovação do Unix e os modernos kernel Linux desenvolveram esta ideia ainda mais. Atualmente, até informação sobre os processos que correm no computador encontra-se no sistema de ficheiros.

Esta representação abstracta e unificada de entidades físicas e processos internos é muito poderosa porque permite-nos utilizar o mesmo comando para o mesmo tipo de operação em muitos dispositivos totalmente diferentes. É mesmo possível alterar o modo como o kernel funciona ao escrever dados em ficheiros especiais que estão ligados a processos em execução.

Dica
Se necessitar identificar a correspondência entre a árvore de ficheiros e a entrada física, execute mount(8) sem argumentos.

1.2.3 Permissões do sistema de ficheiros

As Permissões de sistemas de ficheiros de sistemas tipo-Unix são definidas por três categorias de utilizadores afetados:

- O **utilizador** que é dono do ficheiro (**u**)

- Outros utilizadores no **grupo** ao qual o ficheiro pertence (**g**)
- Todos os **outros** utilizadores (**o**) também referido como ”mundo” e ”todos”

Para o ficheiro, cada permissão correspondente permite as seguintes ações:

- A permissão **read** (**r**) permite ao dono examinar o conteúdo do ficheiro.
- A permissão **write** (**w**) permite ao dono modificar o ficheiro.
- A permissão **execute** (**x**) permite ao dono correr o ficheiro como um comando.

Para o diretório, cada permissão correspondente permite as seguintes ações:

- A permissão **read** (**r**) permite ao dono listar o conteúdo do diretório.
- A permissão **write** (**w**) permite ao dono adicionar ou remover ficheiros no diretório.
- A permissão **execute** (**x**) permite ao dono aceder aos ficheiro no diretório.

Aqui, a permissão **execute** num diretório significa não só permitir a leitura dos ficheiros nesse diretório mas também permitir visualizar os seus atributos, tais como o tamanho e a hora de modificação.

ls(1) é utilizado para mostrar informação de permissões (e mais) para ficheiros e diretórios. Quando é invocado com a opção ”`-l`”, mostra a seguinte informação na ordem apresentada.

- **Tipo de ficheiro** (primeiro caractere)
- **Permissão** de acesso do ficheiro (nove caracteres, a consistir em três caracteres cada para utilizador, grupo e outros por esta ordem)
- **Quantidade de ligações rígidas** ao ficheiro
- Nome do **utilizador** dono do ficheiro
- Nome do **grupo** ao qual o ficheiro pertence
- **Tamanho** do ficheiro em caracteres (bytes)
- **Data e hora** do ficheiro (mtime)
- **Nome** do ficheiro

caractere	**significado**
-	ficheiro normal
d	diretório
l	ligação simbólica
c	nó de dispositivo de caractere
b	nó de dispositivo de bloco
p	named pipe
s	socket

Tabela 1.4: Lista do primeiro caractere da saída de ”`ls -l`”

chown(1) é utilizado a partir da conta de root para alterar o dono do ficheiro. chgrp(1) é utilizado a partir da conta do dono do ficheiro ou da conta root para alterar o grupo do ficheiro. chmod(1) é usado a partir da conta do dono do ficheiro ou da conta root para alterar as permissões de acesso ao ficheiro ou diretório. A sintaxe básica para manipular o ficheiro `foo` é a seguinte.

```
# chown newowner foo
# chgrp newgroup foo
# chmod  [ugoa][+-=][rwxXst][,...] foo
```

Por exemplo, pode fazer com que uma árvore de diretórios tenha como dono o utilizador `foo` e seja partilhada pelo grupo `bar` pelo seguinte:

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

Existem mais três bits especiais de permissões.

- O bit **set user ID** (**s** ou **S** em vez do **x** do utilizador)
- O bit **set group ID** (**s** ou **S** em vez do **x** do grupo)
- O bit **sticky** (**t** ou **T** em vez do **x** dos outros)

Aqui o resultado de "`ls -l`" para estes bits é **capitalizado** se a execução de bits escondidos por estes resultados estiverem **não definidos**.

Definir **set user ID** num ficheiro executável permite a um utilizador executar o ficheiro executável com o ID do dono do ficheiro (por exemplo **root**). De modo semelhante, definir **set group ID** num ficheiro executável permite a um utilizador executar o ficheiro executável com o ID de grupo do ficheiro (por exemplo **root**). Porque estas definições podem causar riscos de segurança, activá-las requer precauções extra.

Definir **set group ID** num diretório activa o esquema de criação de ficheiros ao estilo BSD onde todos os ficheiros criados no diretório pertencem ao **grupo** do diretório.

Definir o **sticky bit** num diretório previne que um ficheiro nesse diretório seja removido por um utilizador que não seja o dono do ficheiro. De modo a tornar o conteúdo de um ficheiro seguro em diretórios onde todos têm acesso de escrita, como o "`/tmp`" ou em diretórios onde um grupo tem acesso de escrita, não basta reiniciar a permissão de **escrita** do ficheiro mas também definir o **sticky bit** no diretório. Caso contrário, o ficheiro pode ser removido e pode ser criado um novo ficheiro com o mesmo nome por qualquer utilizador que tenha acesso de escrita no diretório.

Aqui estão alguns exemplos interessantes de permissões de ficheiros:

```
$ ls -l /etc/passwd /etc/shadow /dev/ppp /usr/sbin/exim4
crw------T 1 root root   108, 0 Oct 16 20:57 /dev/ppp
-rw-r--r-- 1 root root     2761 Aug 30 10:38 /etc/passwd
-rw-r----- 1 root shadow   1695 Aug 30 10:38 /etc/shadow
-rwsr-xr-x 1 root root   973824 Sep 23 20:04 /usr/sbin/exim4
$ ls -ld /tmp /var/tmp /usr/local /var/mail /usr/src
drwxrwxrwt 14 root root  20480 Oct 16 21:25 /tmp
drwxrwsr-x 10 root staff  4096 Sep 29 22:50 /usr/local
drwxr-xr-x 10 root root   4096 Oct 11 00:28 /usr/src
drwxrwsr-x  2 root mail   4096 Oct 15 21:40 /var/mail
drwxrwxrwt  3 root root   4096 Oct 16 21:20 /var/tmp
```

Existe um modo numérico alternativo para descrever as permissões do ficheiro com o chmod(1). Este modo numérico utiliza 3 ou 4 dígitos em numeração octal (radix=8).

Isto parece complicado mas na verdade é bastante simples. Se observar as primeiras colunas (2-10) do resultado do comando "`ls -l`" e lê-las como uma representação binária (radix=2) das permissões do ficheiros ("-" a ser "0" e "rwx" a ser "1"), os últimos três dígitos do valor de modo numérico para si deverão fazer sentido como uma representação octal (radix=8) das permissões do ficheiro.

Por exemplo, tente o seguinte:

digito	significado
1º digito opcional	soma de **set user ID** (=4), **set group ID** (=2) e **sticky bit** (=1)
2º digito	soma das permissões **leitura** (=4), **escrita** (=2) e **executável** (=1) para o **utilizador**
3º digito	idem para **grupo**
4º digito	idem para **outros**

Tabela 1.5: O modo numérico para permissões de ficheiros em comandos chmod(1)

```
$ touch foo bar
$ chmod u=rw,go=r foo
$ chmod 644 bar
$ ls -l foo bar
-rw-r--r-- 1 penguin penguin 0 Oct 16 21:39 bar
-rw-r--r-- 1 penguin penguin 0 Oct 16 21:35 foo
```

Dica
Se necessitar aceder a informação mostrada por "`ls -l`" num script da shell, deve utilizar comandos pertinentes como test(1), stat(1) e readlink(1). Os comandos da própria shell como "`[`" ou "`test`" também podem ser utilizados.

1.2.4 Controlo de permissões para ficheiros acabados de criar: umask

As permissões que são aplicadas ao criar ficheiros e diretórios novos é restringida pelo comando embutido da shell `umask`. Veja dash(1), bash(1) e builtins(7).

```
(file permissions) = (requested file permissions) & ~(umask value)
```

umask	permissões do ficheiro criadas	permissões do diretório criadas	utilização
`0022`	`-rw-r--r--`	`-rwxr-xr-x`	apenas pode ser escrito pelo utilizador
`0002`	`-rw-rw-r--`	`-rwxrwxr-x`	pode ser escrito pelo grupo

Tabela 1.6: Exemplos do valor **umask**

O sistema Debian usa um esquema de grupo privado de utilizadores (UPG). Um UPG é criado sempre que um novo utilizador é adicionado ao sistema. Um UPG tem o mesmo nome que o utilizador para o qual foi criado e esse utilizador é o único membro do UPG. O esquema UPG torna seguro definir a umask para `0002` já que cada utilizador tem o próprio grupo privado dele. (Em algumas variantes de Unix, é bastante comum configurar os utilizadores normais a pertencerem a um único grupo **`users`** e por segurança é uma boa ideia definir a umask para `0022` nesses casos.)

Dica
Active UPG ao pôr "`umask 002`" no ficheiro `~/.bashrc`.

1.2.5 Permissões para grupos de utilizadores (group)



Atenção
Certifique-se de que guarda as alterações não guardadas antes de reiniciar o sistema ou de efetuar ações semelhantes.

Pode adicionar um utilizador `penguin` a um grupo `bird` em dois passos:

- Alterar a configuração do grupo utilizando uma das seguintes opções:
  - Execute "`sudo usermod -aG bird penguin`".
  - Execute "`sudo adduser penguin bird`". (apenas em sistemas Debian típicos)
  - Execute "`sudo vigr`" para `/etc/group` e "`sudo vigr -s`" para `/etc/gshadow` para acrescentar `penguin` na linha para `bird`.
- Aplicar a configuração utilizando uma das seguintes opções:
  - Reiniciar a frio e iniciar sessão. (Melhor opção)
  - Terminar e reiniciar a sessão através do menu GUI. (Isto pode não funcionar sob um Ambiente de Trabalho moderno.)

É possível remover um utilizador `penguin` de um grupo `bird` em dois passos:

- Alterar a configuração do grupo utilizando uma das seguintes opções:
  - Execute "`sudo usermod -rG bird penguin`".
  - Execute "`sudo deluser penguin bird`". (apenas em sistemas Debian típicos)
  - Execute "`sudo vigr`" para `/etc/group` e "`sudo vigr -s`" para `/etc/gshadow` para remover `penguin` na linha para `bird`.
- Aplicar a configuração utilizando uma das seguintes opções:
  - Reiniciar a frio e iniciar sessão. (Melhor opção)
  - Execute "`kill -TERM -1`" e faça algumas ações de correção, como "`systemctl restart NetworkManager.ser`
  - Terminar a sessão através do menu GUI não é uma opção no Gnome Desktop.

Quaisquer tentativas de reinicialização a quente são substitutos frágeis da reinicialização a frio real no sistema de desktop moderno.

Nota
Em alternativa, pode adicionar dinamicamente utilizadores aos grupos durante o processo de autenticação ao adicionar a linha "`auth optional pam_group.so`" a "`/etc/pam.d/common-auth`" e configurar "`/etc/security/group.conf`". (Veja Capítulo 4.)

Os dispositivos de hardware são apenas outro tipo de ficheiros no sistema Debian. Se tiver problemas a aceder a dispositivos como Memórias flash USB e CD-ROM a partir de uma conta de utilizador, deve tornar esse utilizador um membro do grupo relevante.

Alguns grupos notáveis disponibilizados pelo sistema permitem aos seus membros aceder a ficheiros e dispositivos particulares sem privilégios de `root`.

grupo	descrição para ficheiros e dispositivos acessíveis
`dialout`	acesso completo e direto a portas série (”`/dev/ttyS[0-3]`”)
`dip`	Acesso limitado a portas série para ligação **Dialup IP** a peers de confiança
`cdrom`	drives CD-ROM, DVD+/-RW
`audio`	dispositivo de áudio
`video`	dispositivo de vídeo
`scanner`	scanner(es)
`adm`	logs (relatórios) de monitorização do sistema
`staff`	alguns diretórios para trabalho administrativo júnior: ”`/usr/local`”, ”`/home`”

Tabela 1.7: Lista de grupos notáveis disponibilizados pelo sistema para acesso a ficheiros

Dica
Necessita pertencer ao grupo `dialout` para reconfigurar o modem, ligar para qualquer lado, etc. Mas se o `root` criar ficheiros de configuração pré-definidos para peers de confiança em ”`/etc/ppp/peers/`”, apenas precisa de pertencer ao grupo `dip` para criar uma ligação **Dialup IP** para esses peers de confiança a utilizar os comandos pppd(8), pon(1) e poff(1).

Alguns grupos notáveis disponibilizados pelo sistema permitem aos seus membros executar comandos particulares sem privilégios de `root`.

grupo	comandos acessíveis
`sudo`	executa qualquer comando com privilégios de super utilizador
`lpadmin`	executar comandos para adicionar, modificar e remover impressoras das bases de dados de impressoras

Tabela 1.8: Lista de grupos notáveis disponibilizados pelo sistema para execuções de comandos particulares

Para a listagem completa dos utilizadores e grupos disponibilizados pelo sistema, veja a versão recente do documento ”Utilizadores e Grupos” em ”`/usr/share/doc/base-passwd/users-and-groups.html`” disponibilizado pelo pacote `base-passwd`.

Para comandos de gestão para o sistema de utilizador e grupo veja passwd(5), group(5), shadow(5), newgrp(1), vipw(8), vigr(8) e pam_group(8).

1.2.6 Marcas temporais (Timestamps)

Existem três tipos de marcas temporais para um ficheiro de GNU/Linux.

tipo	significado (definição Unix histórica)
mtime	a hora de modificação do ficheiro (`ls -l`)
ctime	a hora de alteração de estado do ficheiro (`ls -lc`)
atime	a hora do último acesso ao ficheiro (`ls -lu`)

Tabela 1.9: Lista dos tipos de marcas temporais

Nota
ctime não é o tempo de criação do ficheiro.

Nota
O valor atual de **atime** num sistema GNU/Linux pode ser diferente daquele da definição Unix histórica.

- Sobrescrever um ficheiro altera todos os atributos **mtime**, **ctime** e **atime** do ficheiro.
- Alterar o dono ou as permissões de um ficheiro altera os atributos **ctime** e **atime** do ficheiro.
- Ler um ficheiro altera o atributo **atime** do ficheiro.no sistema Unix histórico.
- Ler um ficheiro altera o atributo **atime** do ficheiro no sistema GNU/Linux se o sistema de ficheiros dele estiver montado com "`strictatime`".
- Ler um ficheiro pela primeira vez ou após um dia altera o atributo **atime** do ficheiro no sistema GNU/Linux se o sistema de ficheiros dele for montado com `relatime`". (comportamento predefinido desde Linux 2.6.30)
- Ler um ficheiro não altera o atributo **atime** do ficheiro no sistema GNU/Linux se o sistema de ficheiros dele for montado com "`noatime`".

Nota
As opções de montagem "`noatime`" e "`relatime`" são introduzidas para melhorar a performance de leitura do sistema de ficheiros sob casos de utilização normal. Operações simples de leitura de ficheiros sob a opção "`strictatime`" acompanha a operação de escrita que consome tempo para atualizar o atributo **atime**. Mas o atributo **atime** é raramente usado excepto para ficheiro mbox(5). Veja mount(8).

Utilize o comando touch(1) para alterar as marcas temporais de ficheiros existentes.

Para representações de data/hora, o comando `ls` produz cadeias de caracteres localizadas à região configurada que não seja o inglês ("`fr_FR.UTF-8`").

```
$ LANG=C  ls -l foo
-rw-rw-r-- 1 penguin penguin 0 Oct 16 21:35 foo
$ LANG=en_US.UTF-8  ls -l foo
-rw-rw-r-- 1 penguin penguin 0 Oct 16 21:35 foo
$ LANG=fr_FR.UTF-8  ls -l foo
-rw-rw-r-- 1 penguin penguin 0 oct. 16 21:35 foo
```

Dica
Veja Secção 9.3.4 para personalizar a saída do "`ls -l`".

1.2.7 Links (ligações)

Existem dois métodos de associar um ficheiro "`foo`" com um nome de ficheiro diferente "`bar`".

- Hard link
  - Duplicar nome para um ficheiro existente
  - "`ln foo bar`"
- Ligação simbólica ou symlink
  - Ficheiro especial que aponta para outro ficheiro pelo nome
  - "`ln -s foo bar`"

Veja o seguinte exemplo para alterações nas contagens da ligação e as diferenças subtis nos resultados do comando `rm`.

```
$ umask 002
$ echo "Original Content" > foo
$ ls -li foo
1449840 -rw-rw-r-- 1 penguin penguin 17 Oct 16 21:42 foo
$ ln foo bar     # hard link
$ ln -s foo baz  # symlink
$ ls -li foo bar baz
1449840 -rw-rw-r-- 2 penguin penguin 17 Oct 16 21:42 bar
1450180 lrwxrwxrwx 1 penguin penguin  3 Oct 16 21:47 baz -> foo
1449840 -rw-rw-r-- 2 penguin penguin 17 Oct 16 21:42 foo
$ rm foo
$ echo "New Content" > foo
$ ls -li foo bar baz
1449840 -rw-rw-r-- 1 penguin penguin 17 Oct 16 21:42 bar
1450180 lrwxrwxrwx 1 penguin penguin  3 Oct 16 21:47 baz -> foo
1450183 -rw-rw-r-- 1 penguin penguin 12 Oct 16 21:48 foo
$ cat bar
Original Content
$ cat baz
New Content
```

A ligação rígida pode ser feita dentro do mesmo sistema de ficheiros e partilhar o mesmo número de inode tal como o ls(1) com a opção "`-i`" revela.

A ligação simbólica tem sempre permissões nominais de acesso ao ficheiro "`rwxrwxrwx`", conforme mostrado no exemplo acima, com as permissões de acesso efectivas ditadas pelas permissões do ficheiro para o qual aponta.



Cuidado
Geralmente é boa ideia, de todo, não criar ligações simbólicas complicadas ou ligação rígidas a menos que tenha uma boa razão. Podem causar pesadelos onde a combinação lógica das ligações simbólicas resulta em círculos viciosos no sistema de ficheiros.

Nota
Geralmente é preferível utilizar ligações simbólicas em vez de ligação rígidas, a menos que tenha boas razões para usar uma ligação rígida.

O diretório "`.`" liga ao diretório onde ele aparece, assim a contagem de ligações de qualquer novo diretório começa em 2. O diretório "`..`" liga ao diretório pai, assim a contagem de ligações do diretório aumenta com a adição de novos sub-diretórios.

Se está a mudar do Windows para Linux, em breve irá ficar claro o quão bem desenhado está a ligação de nomes de ficheiros em Unix, comparada com o equivalente mais próximo em Windows de "atalhos". Devido a estar implementado no sistema de ficheiros, aplicações não conseguem ver nenhuma diferença entre um ficheiro ligado e o original. No caso de ligações rígidas, não há realmente nenhuma diferença.

1.2.8 Pipes com nome (FIFOs)

Um pipe com nome é um ficheiro que age como um pipe. Coloca algo no ficheiro e sai pelo outro lado. Por isso é chamado um FIFO, ou Primeiro-a-Entrar-Primeiro-a-Sair: a primeira coisa que pôe no pipe é a primeira coisa a sair pelo outro lado.

Se escrever para um pipe com nome, o processo que escreve à pipe não termina até que a informação que está a ser escrita para o pipe seja lida a partir do pipe. Se ler de um pipe com nome, o processo que lê espera até que não

haja mais nada para ler antes de terminar. O tamanho do pipe é sempre zero -- não armazena dados, apenas faz a ligação entre dois processos como a funcionalidade oferecida pelo "|" na sintaxe da shell. No entanto, como este pipe tem um nome, os dois processos não têm de estar na mesma linha de comando ou mesmo serem executados pelo mesmo utilizador. Os pipes foram uma inovação de muita influência do Unix.

Por exemplo, tente o seguinte:

```
$ cd; mkfifo mypipe
$ echo "hello" >mypipe & # put into background
[1] 8022
$ ls -l mypipe
prw-rw-r-- 1 penguin penguin 0 Oct 16 21:49 mypipe
$ cat mypipe
hello
[1]+  Done                    echo "hello" >mypipe
$ ls mypipe
mypipe
$ rm mypipe
```

1.2.9 Sockets

Os sockets são usados extensivamente por toda a comunicação da Internet, bases de dados e pelo próprio sistema operativo. São semelhantes a pipes com nome (FIFO) e permitem aos processos trocarem informação mesmo entre computadores diferentes. Para o socket, esses processos não precisam de estar a correr ao mesmo tempo, nem correrem como filhos do mesmo processo pai. Isto é o destino da comunicação inter-processo (IPC). A troca de informação pode ocorrer sobre a rede entre máquinas diferentes. Os dois mais comuns são o socket de Internet e o socket de domínio Unix.

Dica
"`ss -t -u -a`" (do pacote `iproute2`) dá uma visão geral, muito útil, dos sockets que estão abertos num determinado sistema.

1.2.10 Ficheiros de dispositivo

Os ficheiros de Dispositivos referem-se a dispositivos físicos ou virtuais no seu sistema, como o seu disco rígido, placa gráfica, monitor ou teclado. Um exemplo de dispositivo virtual é a consola, que é representada por "`/dev/console`".

Existem 2 tipos de ficheiros de dispositivo.

- **Dispositivo de Caractere**
  - Acedido por um caractere de cada vez
  - 1 caractere = 1 byte
  - Por exemplo, dispositivo de teclado, porta serial, …
- **Dispositivo de Bloco**
  - acedido em unidades maiores chamadas blocos
  - 1 bloco > 1 byte
  - Por exemplo, o disco rígido, …

Pode ler e escrever nos ficheiros de dispositivo, embora o ficheiro possa muito bem conter dados binários que podem ser uma salada incompreensível para humanos. Escrever dados directamente nestes ficheiros é por vezes útil para diagnosticar problemas com ligações de hardware. Por exemplo, pode despejar um ficheiro de texto para um dispositivo de impressora "`/dev/lp0`" ou enviar comandos de modem à porta serial apropriada "`/dev/ttyS0`". Mas, a menos que isto seja feito com cuidado, pode causar problemas maiores. Portanto seja cauteloso.

Nota
Para o acesso normal a uma impressora, use lp(1).

Os números de nós de dispositivo são mostrados ao executar ls(1) como a seguir.

```
$ ls -l /dev/sda /dev/sr0 /dev/ttyS0 /dev/zero
brw-rw---T  1 root disk     8,  0 Oct 16 20:57 /dev/sda
brw-rw---T+ 1 root cdrom   11,  0 Oct 16 21:53 /dev/sr0
crw-rw---T  1 root dialout  4, 64 Oct 16 20:57 /dev/ttyS0
crw-rw-rw-  1 root root     1,  5 Oct 16 20:57 /dev/zero
```

- "`/dev/sda`" tem o número maior de dispositivo 8 e o número menor de dispositivo 0. Isto é acessível para leitura e escrita aos utilizadores que pertencem ao grupo `disk`.
- "`/dev/sr0`" tem o número maior de dispositivo 11 e o número menor de dispositivo 0. Isto é acessível para leitura e escrita aos utilizadores que pertencem ao grupo `cdrom`.
- "`/dev/ttyS0`" tem o número maior de dispositivo 4 e o número menor de dispositivo 64. Isto é acessível para leitura e escrita aos utilizadores que pertencem ao grupo `dialout`.
- "`/dev/zero`" tem o número 1 no dispositivo maior e o número 5 no número de dispositivo menor. Isto é acessível para leitura/escrita a todos.

No sistema Linux moderno, o sistema de ficheiros sob "`/dev/`" é povoado automaticamente pelo mecanismo udev(7).

1.2.11 Ficheiros de dispositivos especiais

Existem alguns ficheiros de dispositivos especiais.

ficheiro de dispositivo	acção	descrição da resposta
`/dev/null`	ler	retorna o "caractere de fim-de-ficheiro (EOF)"
`/dev/null`	escrever	retorna nada (um poço de despejo de dados sem fundo)
`/dev/zero`	ler	retorna "o caractere `\0` (NULO)" (não é o mesmo que o número zero em ASCII)
`/dev/random`	ler	retorna caracteres aleatórios a partir de um verdadeiro gerador de números aleatórios, a entregar entropia real (lento)
`/dev/urandom`	ler	retorna caracteres aleatórios a partir de um gerador de números pseudo-aleatórios criptograficamente seguros
`/dev/full`	escrever	retorna o erro de disco cheio (ENOSPC)

Tabela 1.10: Lista de ficheiros de dispositivos especiais

Estes são usados frequentemente em conjunto com o redireccionamento da shell (veja Secção 1.5.8).

1.2.12 procfs e sysfs

Os procfs e sysfs montados em "`/proc`" e "`/sys`" são os pseudo-sistemas-de-ficheiros e expõem estruturas de dados internas do kernel ao espaço de utilizador. Por outras palavras, estas entradas são virtuais, a significar que elas agem como uma janela de conveniência às operações do sistema operativo.

O diretório "`/proc`" contém (entre outras coisas) um sub-diretório para cada processo em execução no sistema, o qual tem o nome do ID do processo (PID). Os utilitários do sistema que acedem à informação de processos, como o ps(1), obtêm a informação deles de esta estrutura de diretórios.

Os diretórios sob "`/proc/sys/`" contêm interfaces para alterar certos parâmetros do kernel durante o funcionamento. (Pode fazer o mesmo através do comando especializado sysctl(8) ou do ficheiro de configuração/pré-carregamento dele "`/etc/sysctl.d/*.conf`".)

As pessoas frequentemente assustam-se quando notam num ficheiro em particular - "`/proc/kcore`" - o qual é geralmente enorme. Isto é (mais ou menos) uma cópia do conteúdo da memória do seu computador. É usado para depuração do kernel. É um ficheiro virtual que aponta à memória do computador, portanto não se preocupe com o tamanho dele.

O diretório em "`/sys`" contém estruturas de dados do kernel exportadas, os seus atributos e as suas ligações entre eles. Também contém interfaces para alterar certos parâmetros do kernel durante o funcionamento.

Veja "`proc.txt(.gz)`", "`sysfs.txt(.gz)`" e outros documentos relacionados na documentação do kernel Linux ("`/usr/share/doc/linux-doc-*/Documentation/filesystems/*`") disponibilizados pelo pacote `linux-doc-*`.

1.2.13 tmpfs

O tmpfs é um sistema de ficheiros temporário o qual mantém todos os ficheiros na memória virtual. Os dados de tmpfs na cache de páginas na memória pode ser ir ao espaço swap no disco, conforme necessário.

O diretório "`/run`" é montado como tmpfs no inicio do processo de arranque. Isto ativa a escrita mesmo que o diretório "`/` esteja montado como apenas-leitura. Esta é a nova localização para o armazenamento de ficheiros de estado transitório e substitui várias outras localizações descritas na Filesystem Hierarchy Standard, versão 2.3:

- "`/var/run`" → "`/run`"
- "`/var/lock`" → "`/run/lock`"
- "`/dev/shm`" → "`/run/shm`"

Veja "`tmpfs.txt(.gz)`" na documentação do kernel Linux ("`/usr/share/doc/linux-doc-*/Documentation/files`" disponibilizada pelo pacote `linux-doc-*`.

1.3 Midnight Commander (MC)

Midnight Commander (MC) é um "Canivete Suíço" GNU para a consola Linux e para outros ambientes de terminal. Isto oferece aos novatos uma experiência de consola movida por menus o que é mais fácil de aprender do que os comandos standard do Unix.

Pode necessitar instalar o pacote Midnight Commander que é intitulado de "`mc`" com o seguinte:

```
$ sudo apt-get install mc
```

Utilize o comando mc(1) para explorar o sistema Debian. Esta é a melhor maneira de aprender. Por favor explore algumas localizações interessantes apenas a usar as teclas do cursor e Enter.

- "`/etc`" e os seus sub-diretórios
- e"`/var/log`" os seus sub-diretórios
- "`/usr/share/doc`" os seus sub-diretórios
- "`/usr/sbin`" e "`/usr/bin`"

1.3.1 Personalização do MC

De modo a fazer o MC mudar o diretório de trabalho ao sair e `cd` para o diretório, sugiro modificar o "`~/.bashrc`" para incluir um script disponibilizado pelo pacote `mc`.

```
. /usr/lib/mc/mc.sh
```

Veja mc(1) (sob a opção "`-P`") para a razão. (Se não compreende exatamente do que estou a falar aqui, pode fazer isto mais tarde.)

1.3.2 Iniciar o MC

O MC pode ser iniciado com o seguinte:

```
$ mc
```

O MC toma conta de operações de ficheiros através do menu dele, a requerer esforços mínimos do utilizador. Carregue em F1 para obter ajuda. Pode brincar com o MC apenas com as teclas de cursor e teclas de função.

Nota
Em algumas consolas, como o gnome-terminal(1), as teclas de função podem ser roubadas pelo programa da consola. Pode desativar estas funcionalidades no menu "Preferências" → "Geral" e "Atalhos" do `gnome-terminal`.

Se encontrar problemas com a codificação de caracteres que mostram caracteres com lixo, adicionar "`-a`" à linha de comandos do MC pode ajudar a prevenir problemas.

Se isto não limpar os seus problemas no ecrã com o MC, veja Secção 8.3.1.

1.3.3 Gestor de ficheiros no MC

O predefinido são dois painéis de diretórios que contêm listas de ficheiros. Outro modo útil é definir a janela da direita para "informação" para ver informações de privilégios de acesso a ficheiros, etc. De seguida existem algumas teclas de atalho essenciais. Com o daemon gpm(8) em execução, também se pode usar um rato em consolas de caracteres no Linux. (Certifique-se que carrega na tecla shift para obter o comportamento normal de cortar e colar no MC.)

tecla	**tecla de atalho**
`F1`	menu de ajuda
`F3`	visualizador interno de ficheiros
`F4`	editor interno
`F9`	ativar o menu de desenrolar
`F10`	sair do Midnight Commander
`Tab`	mover entre duas janelas
`Insert` ou `Ctrl-T`	marcar o ficheiro para uma operação de múltiplos ficheiros como uma cópia
`Del`	apagar o ficheiro (tenha cuidado -- configure o MC para modo de apagar seguro)
Teclas do cursor	auto-explicativo

Tabela 1.11: As teclas de atalho do MC

1.3.4 Truques de linha de comandos no MC

- O comando `cd` altera o diretório mostrado no ecrã selcionado.
- `Ctrl-Enter` ou `Alt-Enter` copia um nome de ficheiro para a linha de comandos. Utilize isto com os comandos cp(1) e mv(1) em conjunto com a edição de linha de comandos.
- `Alt-Tab` mostra escolhas de expansão de nomes de ficheiros da shell.
- Pode-se especificar os diretórios de inicio para ambas janelas como argumentos ao MC; por exemplo, "`mc /etc /root`".
- `Esc` + `n-key` → `Fn` (isto é, `Esc` + `1` → `F1`, etc.; `Esc` + `0` → `F10`)
- Carregar em `Esc` antes da tecla tem o mesmo efeito que carregar em `Alt` e na tecla em conjunto; isto é, carregar em `Esc` + `c` para `Alt-C`. `Esc` é chamada uma meta-tecla e por vezes mencionada como "`M-`".

1.3.5 O editor interno em MC

O editor interno tem um esquema de cortar-e-colar interessante. Pressionar `F3` marca o inicio da seleção, um segundo `F3` marca o final da seleção e destaca a seleção. Depois pode mover o cursor. Se pressionar F6, a área selcionada é movida à localização do cursor. Se pressionar F5, a área selcionada é copiada e inserida na localização do cursor. `F2` salva o ficheiro. `F10` fá-lo sair. A maioria das teclas do cursor funcionam de modo intuitivo.

Este editor pode ser iniciado directamente num ficheiro a usar um dos seguintes comandos.

```
$ mc -e filename_to_edit
```

```
$ mcedit filename_to_edit
```

Este não é um editor de várias janelas, mas podem-se utilizar várias consolas de Linux para se conseguir o mesmo efeito. Para copiar entre janelas, utilize as teclas Alt-F*n* para mudar entre consolas virtuais utilize "Ficheiro→Inserir ficheiro" ou "Ficheiro→Copiar para ficheiro" para mover uma porção de um ficheiro para outro ficheiro.

Este editor interno pode ser substituído por qualquer editor externo à escolha.

Além disso, muitos programas utilizam as variáveis de ambiente "`$EDITOR`" ou "`$VISUAL`" para decidir que editor utilizar. Se não está confortável com o vim(1) ou com o nano(1), pode definir a "`mcedit`" a adicionar as seguintes linhas ao "`~/.bashrc`".

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

Recomendo definir isto para "`vim`" se possível.

Se não está confortável com o vim(1), pode continuar a utilizar o mcedit(1) para a maioria das tarefas de manutenção do sistema.

1.3.6 O visualizador interno no MC

MC é um visualizador muito inteligente. Esta é uma grande ferramenta para procurar palavras em documentos. Uso sempre isto para ficheiros no diretório "`/usr/share/doc`". Esta é a maneira mais rápida para navegar por entre grandes quantidades de informação de Linux. Este visualizador pode ser iniciado diretamente a usar um dos seguintes comandos:

```
$ mc -v path/to/filename_to_view
```

```
$ mcview path/to/filename_to_view
```

1.3.7 Funcionalidades de auto-arranque do MC

Carregue em Enter num ficheiro e o programa apropriado lida com o conteúdo do ficheiro (veja Secção 9.4.11). Esta é uma funcionalidade muito conveniente do MC.

tipo de ficheiro	**reacção à tecla enter**
ficheiro executável	executa comando
ficheiro man	canaliza o conteúdo para software de visualização
ficheiro html	canaliza o conteúdo para explorador web
ficheiros "`*.tar.gz`" e "`*.deb`"	explora o conteúdo dele como se fosse um sub-diretório

Tabela 1.12: A reacção à tecla enter no MC

De modo a permitir esta visualização e funcionalidades de ficheiros virtuais , os ficheiros a visualizar não devem ser definidos como executáveis. Mude o estado deles com o chmod(1) ou via menu ficheiro do MC.

1.3.8 Sistema de ficheiros virtual do MC

MC pode ser utilizado para aceder a ficheiros através da Internet. Vá ao menu premindo F9, "`Enter`" e "h" para ativar o sistema de ficheiros Shell. Introduza um URL na forma "`sh://[user@]machine[:options]/[remote-dir]`", que recupera um diretório remoto que aparece como um diretório local utilizando o `ssh`.

1.4 O ambiente de trabalho estilo Unix básico

Apesar do MC lhe permitir fazer quase tudo, é muito importante aprender a utilizar as ferramentas de linha de comandos invocadas a partir da prompt da shell e familiarizar-se com o ambiente de trabalho do tipo Unix.

1.4.1 A shell de login

Como a shell de inicio de sessão pode ser usada por alguns programas de inicialização do sistema, é prudente mantê-la como bash(1) e evitar mudar a shell de inicio de sessão com chsh(1).

Se quiser utilizar uma linha de comandos interativa diferente, defina-a a partir da configuração do emulador de terminal GUI ou inicie-a a partir de `~/.bashrc`, e.g., colocando aí "`exec /usr/bin/zsh -i -l`" ou "`exec /usr/bin/fish -i -l`".

Dica
Apesar das shells tipo POSIX partilharem a sintaxe básica, podem diferir no comportamento em coisas tão básicas como variáveis de shell e expansões glob. Por favor verifique as suas documentações para detalhes.

Neste capítulo do tutorial, a shell interactiva significa sempre `bash`.

1.4.2 Personalizar bash

Pode personalizar o comportamento da bash(1) em "`~/.bashrc`".

Por exemplo, tente o seguinte.

pacote	popcon	tamanho	Shell do POSIX	descrição
bash	V:886, I:999	7308	Sim	Bash: a GNU Bourne Again SHell (o standard de facto)
bash-completion	V:37, I:962	1952	N/D	conclusão programável para a shell bash
dash	V:929, I:999	207	Sim	Shell Almquist, bom para scripts da shell
zsh	V:39, I:68	2571	Sim	Z shell: a shell standard com muitas melhorias
tcsh	V:4, I:14	1366	Não	Shell TENEX C: uma versão melhorada de Berkeley csh
mksh	V:4.7, I:7.5	7713	Sim	Uma versão de Korn shell
csh	V:1.0, I:5.2	348	Não	OpenBSD shell C, uma versão do Berkeley csh
sash	V:0.3, I:4.7	1245	Sim	shell Stand-alone com comandos embutidos (Não se destina a "/usr/bin/sh" standard)
ksh	V:0.3, I:7.9	65	Sim	a real, versão AT&T da Korn shell
rc	V:0.07, I:0.70	182	Não	implementação da AT&T Plan 9 rc shell
posh	V:0.02, I:0.22	195	Sim	Policy-compliant Ordinary SHell (deriva da pdksh)

Tabela 1.13: Lista de programas da shell

```
# enable bash-completion
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

# CD upon exiting MC
. /usr/lib/mc/mc.sh

# set CDPATH to a good one
CDPATH=.:/usr/share/doc:~:~/Desktop:~
export CDPATH

PATH="${PATH+$PATH:}/usr/sbin:/sbin"
# set PATH so it includes user's private bin if it exists
if [ -d ~/bin ] ; then
  PATH="~/bin${PATH+:$PATH}"
fi
export PATH

EDITOR=vim
export EDITOR
```

Dica
Pode encontrar mais dicas de personalização da bash, como os Secção 9.3.6, em Capítulo 9.

Dica
O pacote bash-completion activa preenchimento automático programável para bash.

1.4.3 Teclas especiais

No ambiente do tipo Unix, existem algumas combinações de teclas que têm significados especiais. Por favor note que numa consola de caracteres normal do Linux, apenas as teclas `Ctrl` e `Alt` do lado esquerdo funcionam como se espera. Aqui estão algumas combinações de teclas notáveis para se lembrar.

tecla	descrição do atalho da tecla
`Ctrl-U`	apagar a linha antes do cursor
`Ctrl-H`	apagar um caractere antes do cursor
`Ctrl-D`	termina a entrada (sai da shell se estiver a usar uma shell)
`Ctrl-C`	termina um programa em funcionamento
`Ctrl-Z`	pára temporariamente o programa ao movê-lo para segundo plano
`Ctrl-S`	pára a saída ao ecrã
`Ctrl-Q`	reactiva a saída para o ecrã
`Ctrl-Alt-Del`	reinicia/pára o sistema, veja inittab(5)
`Left-Alt-key` (opcionalmente, `Windows-key`)	meta-tecla para o Emacs e a UI semelhante
`Up-arrow`	inicia a busca no histórico de comandos em `bash`
`Ctrl-R`	inicia o histórico de comandos incremental em `bash`
`Tab`	completa a entrada do nome de ficheiro para a linha de comandos em `bash`
`Ctrl-V Tab`	entrada `Tab` sem expansão para a linha de comandos em `bash`

Tabela 1.14: Lista de teclas de atalho para bash

Dica
A funcionalidade do terminal de `Ctrl-S` pode ser desactivada a usar stty(1).

1.4.4 Operações com o rato

Operações de rato para texto no sistema Debian misturam 2 estilos com algumas variações:

- Operações tradicionais do rato ao estilo Unix:
  - utilizar 3 botões (clicar)
  - usar PRIMÁRIA
  - utilizado por aplicações X como o `xterm` e aplicações de texto na consola Linux
- Operações do rato ao estilo GUI moderno:
  - utilizar 2 botões (arrastar + clicar)
  - utilizar PRIMÁRIA e ÁREA DE TRANSFERÊNCIA
  - utilizado em aplicações GUI modernas, como o `gnome-terminal`

Aqui, a seleção PRIMÁRIA é o intervalo de texto realçado. Dentro do programa de terminal, `Shift-Ctrl-C` é utilizado em vez disso para evitar terminar um programa em execução.

A roda central nos modernos ratos de roda é considerada o botão central do rato e pode ser usada como terceiro botão ou clique-central. Clicar nos botões esquerdo e direito ao mesmo tempo serve como clique-central em situações de sistema de rato de 2 botões.

De modo a utilizar um rato nas consolas de caracteres Linux, é necessário ter o gpm(8) a correr como daemon.

acção	**resposta**
Clique-esquerdo-e-arrastar do rato	selecionar intervalo como seleção PRIMÁRIA
Clique-esquerdo	selecionar o início do intervalo para a seleção PRIMÁRIA
Clique com botão direito (tradicional)	selecionar o fim do intervalo para a seleção PRIMÁRIA
Clique com botão direito (moderno)	menu dependente do contexto (cortar/copiar/colar)
Clique com o botão do meio ou Shift-`Ins`	inserir seleção PRIMÁRIA no cursor
`Ctrl-X`	cortar a seleção PRIMÁRIA para a ÁREA DE TRANSFERÊNCIA
`Ctrl-C` (`Shift-Ctrl-C` no terminal)	copia a seleção PRIMÁRIA para a ÁREA DE TRANSFERÊNCIA
`Ctrl-V`	colar ÁREA DE TRANSFERÊNCIA no cursor

Tabela 1.15: Lista de operações do rato e ações-chave relacionadas no Debian

1.4.5 O pager

O comando less(1) é o paginador avançado (navegador de conteúdo de ficheiros). Ele lê o ficheiro especificado pelo seu argumento de comando ou pela sua entrada standard. Carregue em "h" se precisar de ajuda enquanto explora com o comando `less`. Pode fazer muito mais do que o more(1) e pode ser ampliado ao executar "`eval $(lesspipe)`" ou "`eval $(lessfile)`" no script de arranque de shell. Veja mais em "`/usr/share/doc/less/LESSOP`". A opção "`-R`" permite saída em caracteres em bruto e activa sequências de escape de cores ANSI. Veja less(1).

Dica
No comando `less`, escreva "h" para ver o ecrã de ajuda, escreva "`/`" ou "?" para procurar uma cadeia de caracteres e escreva "`-i`" para alterar a sensibilidade às maiúsculas e minúsculas.

1.4.6 O editor de texto

Deve tornar-se conhecedor de uma das variantes dos programas Vim ou Emacs que são populares em sistemas tipo Unix.

Acho que habituar-se aos comandos do Vim é a coisa certa a fazer, pois o editor Vim está sempre presente no mundo Linux/Unix. (Na verdade, o `vi` original ou o novo `nvi` são programas que vai encontrar em todo o lado. Escolho o Vim para novatos porque é bastante semelhante e mais poderoso já que lhe oferece ajuda através da tecla `F1` .)

Pode escolher o Emacs ou o XEmacs como o seu editor favorito, que é realmente uma outra boa escolha, particularmente para a programação. O Emacs tem também um leque de outras funcionalidades, incluindo funcionar como um leitor de news, editor de diretório, programa de mail, etc. Quando é usado para programação ou edição de scripts de shell, é inteligente para reconhecer o formato daquilo em que está a trabalhar e tenta disponibilizar assistência. Algumas pessoas afirmam que o único programa que necessitam em Linux é o Emacs. 10 minutos a aprender Emacs agora pode poupar-lhe horas mais tarde. Ter o manual do Emacs GNU para referência quando se aprende Emacs é altamente recomendado.

Todos estes programas vêm normalmente com um programa tutor para queaprenda a usá-los pela prática. Arranque o Vim ao escrever "`vim`" e carregue an tecla F1. Deverá ler pelo menos as primeiras 35 linhas. Depois faça o curso de treino online ao mover o cursor para "`|tutor|`" e pressionar `Ctrl-]`.

Nota
Os bons editores, como o Vim e o Emacs, podem lidar corretamente com UTF-8 e outros textos com codificações exóticas. É uma boa ideia usar o ambiente GUI com o locale UTF-8 e instalar as fontes e programas necessários para tal. Os editores têm opções para definir a codificação do ficheiro independentemente do ambiente GUI. Por favor consulte a documentação acerca de texto multibyte.

1.4.7 Definir um editor de texto predefinido

Debian vem com uma quantia de editores diferentes. Recomendamos instalar o pacote `vim`, como mencionado anteriormente.

Debian disponibiliza acesso unificado ao editor predefinido do sistema através do comando "`/usr/bin/editor`" para que outros programas (p.e., reportbug(1)) possam invocá-lo. Pode alterá-lo com o seguinte.

```
$ sudo update-alternatives --config editor
```

A escolha de "`/usr/bin/vim.basic`" em vez de "`/usr/bin/vim.tiny`" é uma recomendação minha para novatos pois suporta destaque de sintaxe.

Dica
Muitos programas utilizam as variáveis de ambiente "`$EDITOR`" ou "`$VISUAL`" para decidir qual o editor que vai utilizar (veja Secção 1.3.5 e Secção 9.4.11). Para consistência no sistema Debian, defina estas para "`/usr/bin/editor`". (Historicamente, "`$EDITOR`" era "`ed`" e "`$VISUAL`" era "`vi`".)

1.4.8 Utilizando o vim

O recente vim(1) inicia-se na opção sã "`nocompatible`" e entra no modo `NORMAL`.[1]

modo	**toques de tecla**	**acção**
`NORMAL`	`:help|only`	apresentar o ficheiro de ajuda
`NORMAL`	`:e filename.ext`	abre uma nova memória intermédia para editar o ficheiro `filename.ext`
`NORMAL`	`:w`	substituir a memória intermédia atual pelo ficheiro original
`NORMAL`	`:w filename.ext`	escreve a memória intermédia atual em `filename.ext`
`NORMAL`	`:q`	sair do `vim`
`NORMAL`	`:q!`	forçar a saída do `vim`
`NORMAL`	`:only`	fechar todas as outras janelas abertas divididas
`NORMAL`	`:set nocompatible?`	verificar se `vim` está no modo "são" `nocompatible`
`NORMAL`	`:set nocompatible`	definir o `vim` para modo limpo `nocompatible`
`NORMAL`	`i`	entrar no modo `INSERT`
`NORMAL`	`R`	entrar no modo `REPLACE`
`NORMAL`	`v`	entrar no modo `VISUAL`
`NORMAL`	`V`	entrar no modo `VISUAL` em linha
`NORMAL`	Ctrl-V	entrar no modo `VISUAL` em bloco
exceto `TERMINAL-JOB`	Tecla ESC	entrar no modo `NORMAL`
`NORMAL`	`:term`	entrar no modo `TERMINAL-JOB`
`TERMINAL-NORMAL`	`i`	entrar no modo `TERMINAL-JOB`
`TERMINAL-JOB`	Ctrl-`W N` (ou Ctrl-\ Ctrl-`N`)	entrar no modo `TERMINAL-NORMAL`
`TERMINAL-JOB`	Ctrl-`W :`	entre no modo `Ex`-mode no `TERMINAL-NORMAL`

Tabela 1.16: Lista de teclas básicas do Vim

Por favor, use o programa "`vimtutor`" para aprender `vim` através de um curso tutorial interativo.

1Mesmo o `vim` mais antigo pode iniciar no modo "`nocompatible`" (não compatível), iniciando-o com a opção "`-N`".

O programa `vim` muda o seu comportamento em relação às teclas digitadas com base no **modo**. A digitação de teclas para buffer é feita principalmente nos modos `INSERT` e `REPLACE`. A deslocação do cursor é feita principalmente no modo `NORMAL`. A seleção interativa é feita no modo `VISUAL`. Escrever "`:`" no modo `NORMAL` muda o seu **modo** para o modo `Ex`. O modo `Ex` aceita comandos.

Dica
O Vim vem com o pacote **Netrw**. O Netrw suporta a leitura de ficheiros, a escrita de ficheiros, a navegação em diretórios através de uma rede e a navegação local! Experimente o Netrw com "`vim .`" (um ponto como argumento) e leia o seu manual em "`:help netrw`".

Para a configuração avançada do `vim`, consulte Secção 9.2.

1.4.9 Gravar as atividades da shell

O resultado do comando na shell pode sair fora do seu ecrã e ficar perdido para sempre. É uma boa prática registar as atividades da shell num ficheiro para rever mais tarde. Este tipo de registo é essencial quando executa quaisquer tarefas de administração do sistema.

Dica
O novo Vim (versão>=8.2) pode ser usado para gravar as atividades da shell de forma limpa usando o modo `TERMINAL-JOB`. Ver Secção 1.4.8.

O método básico de gravar a atividade da shell é corrê-la sob script(1).

Por exemplo, tente o seguinte:

```
$ script
Script started, file is typescript
```

Faz quaisquer comandos de shell sob `script`.

Carregue em `Ctrl-D` para terminar o `script`.

```
$ vim typescript
```

Veja Secção 9.1.1 .

1.4.10 Comandos básicos de Unix

Vamos aprender comandos básicos do Unix. Aqui Uso "Unix" no sentido genérico dele. Geralmente qualquer SO clone do Unix oferece comandos equivalentes. O sistema Debian não é excepção. Não se preocupe se alguns comandos não funcionarem como deseja por agora. Se for utilizado `alias` na shell, as saídas dos comandos correspondentes serão diferentes. Estes exemplos não se destinam a ser executados por esta ordem.

Tente os seguintes comandos a partir da conta de utilizador não-privilegiado.

comando	descrição
`pwd`	mostrar o nome do diretório atual
`whoami`	mostrar o nome do utilizador atual
`id`	mostrar a identidade do utilizador atual (nome, uid, gid e grupos associados)
`file foo`	mostrar o tipo de ficheiro para o ficheiro "*foo*"
`type -p commandname`	mostrar a localização de um ficheiro do comando "*commandname*"
`which commandname`	, ,
`type commandname`	mostrar informação do comando "*commandname*"
`apropos key-word`	encontra comandos relacionados com a "*key-word*"
`man -k key-word`	, ,
`whatis commandname`	mostra explicação de uma linha para o comando "*commandname*"
`man -a commandname`	mostrar a explicação do comando "*commandname*" (estilo Unix)
`info commandname`	mostrar uma explicação longa do comando "*commandname*" (estilo GNU)
`ls`	listar o conteúdo do diretório (ficheiros e diretórios não escondidos)
`ls -a`	listar o conteúdo do diretório (todos os ficheiros e diretórios)
`ls -A`	listar o conteúdo do diretório (quase todos os ficheiros e diretórios, isto é, salta o "`..`" e "`.`")
`ls -la`	listar todo o conteúdo do diretório com informação detalhada
`ls -lai`	listar todo o conteúdo do diretório com número de inode e informação detalhada
`ls -d`	listar todos os diretórios sob o diretório atual
`tree`	mostrar o conteúdo da árvore de ficheiros
`lsof foo`	listar o estado aberto do ficheiro "*foo*"
`lsof -p pid`	listar ficheiros abertos pelo processo de ID: "*pid*"
`mkdir foo`	criar um novo diretório "*foo*" no diretório atual
`rmdir foo`	remover um diretório "*foo*" no diretório atual
`cd foo`	mudar o diretório para o diretório "*foo*" no diretório atual ou no diretório listado na variável "`$CDPATH`"
`cd /`	mudar o diretório para o diretório raiz
`cd`	mudar ao diretório home do utilizador atual
`cd /foo`	mudar para o diretório de caminho absoluto "*/foo*"
`cd ..`	mudar ao diretório pai
`cd ~foo`	mudar ao diretório home do utilizador "*foo*"
`cd -`	mudar ao diretório anterior
`</etc/motd pager`	mostrar o conteúdo de "`/etc/motd`" a utilizar o paginador predefinido
`touch junkfile`	criar um ficheiro vazio "*junkfile*"
`cp foo bar`	copiar um ficheiro "*foo*" existente para um novo ficheiro "*bar*"
`rm junkfile`	remover um ficheiro "*junkfile*"
`mv foo bar`	renomear um ficheiro "*foo*" existente para um novo nome "*bar*" ("*bar*" não pode existir)
`mv foo bar`	mover um ficheiro "*foo*" existente para uma nova localização *bar/foo* (o diretório "*bar*" tem de existir)
`mv foo bar/baz`	mover um ficheiro existente "*foo*" para uma nova localização com um novo nome "*bar/baz*" (o diretório "*bar*" tem de existir mas o diretório "*bar/baz*" não pode existir)
`chmod 600 foo`	tornar um ficheiro existente "*foo*" proibido de ser lido e ser escrito por outras pessoas (não executável para todos)
`chmod 644 foo`	tornar um ficheiro existente "*foo*" permissível de ser lido mas proibido de ser escrito por outras pessoas (não executável para todos)
`chmod 755 foo`	tornar um ficheiro existente "*foo*" permissível de ser lido mas proibido de ser escrito por outras pessoas (executável para todos)
`find . -name pattern`	procurar nomes de ficheiros correspondentes usando um "*pattern*" de shell (lento)
`locate -d . pattern`	procurar nomes de ficheiros correspondentes usando um "*pattern*" de shell (mais rápido ao usar uma base de dados gerada regularmente)

Nota
Unix tem a tradição de esconder os nomes de ficheiros que começam por ”`.`”. Eles são tradicionalmente ficheiros que contêm informação de configuração e preferências do utilizador.
Para o comando `cd`, veja builtins(7).
O paginador predefinido da vastidão do sistema Debian é o more(1), o qual não pode deslocar para trás. A instalar o pacote `less` a usar o comando ”`apt-get install less`”, o less(1) torna-se o paginador predefinido e pode deslocar para trás com as teclas do cursor.
O ”`[`” e ”`]`” na expressão regular do comando ”`ps aux | grep -e "[e]xim4*"`” em cima activam `grep` para evitar a correspondência consigo próprio. O ”`4*`” na expressão regular significa 0 ou mais repetições do caractere ”4” assim activa o `grep` a corresponder a ambos ”`exim`” e ”`exim4`”. Apesar de ”`*`” ser usado no glob de nome de ficheiro da shell e na expressão regular, os seus significados são diferentes. Aprenda a expressão regular a partir do grep(1).

Por favor percorra os diretórios e espreite no sistema a usar os comandos em cima como treino. Se tiver questões sobre qualquer comando de consola, por favor certifique-se de ler o manual dele.

Por exemplo, tente o seguinte:

```
$ man man
$ man bash
$ man builtins
$ man grep
$ man ls
```

Pode ser um pouco difícil de habituar-se ao estilo dos manuais, porque são bastante concisos, particularmente os mais antigos, muito tradicionais. Mas assim que se habituar a eles, vai apreciar a brevidade deles.

Por favor note que muitos comandos do tipo Unix incluindo os GNU e BSD mostram informação breve de ajuda se os invocar numa das seguintes formas (ou sem argumentos nalguns casos).

```
$ commandname --help
$ commandname -h
```

1.5 O comando simples da shell

Agora tem alguma prática de como utilizar o sistema Debian. Vamos ver mais fundo no mecanismo da execução de comandos no sistema Debian. Aqui simplifiquei a realidade para o novato. Veja bash(1) para a explicação exata.

Um comando simples é uma sequência de componentes.

1. Atribuições de variáveis (opcional)
2. Nome do comando
3. Argumentos (opcional)
4. Re-direcções (opcional: > , >> , < , << , etc.)
5. Operador de controle (opcional: && , | | , *nova-linha* , ; , & , (,))

1.5.1 Execução do comando e variável de ambiente

Os valores de algumas variáveis de ambiente modificam o comportamento de alguns comandos de Unix.

Os valores predefinidos das variáveis de ambiente são definidos inicialmente pelo sistema PAM e depois alguns deles podem ser redefinidos por alguns programas.

- O sistema PAM, como o `pam_env`, pode definir variáveis de ambiente em "`/etc/pam.conf`", "`/etc/environment`" e "`/etc/default/locale`".
- O ambiente GUI moderno corre a partir da unidade systemd de modo de utilizador (veja systemd.unit(5)) como define as suas variáveis de ambiente por um ficheiro no diretório "`~/.config/environment.d/`".
- A inicialização do programa específico do utilizador pode repor variáveis de ambiente através de "`~/.profile`", "`~/.bash_profile`" e "`~/.bashrc`".

1.5.2 A variável "$LANG"

A configuração regional predefinida é definida na variável de ambiente "`$LANG`" e é configurada como "`LANG=xx_YY.UTF-8`" pelo instalador ou pela configuração GUI subsequente, por exemplo, "Definições" → "Região & Idioma" → "Idioma" / "Formatos" para o GNOME.

Nota
Recomendo que configure o ambiente do sistema por agora apenas pela variável "`$LANG`" e se mantenha afastado das variáveis "`$LC_*`" a menos que seja absolutamente necessário.

O valor completo do locale dado à variável "`$LANG`" consiste em 3 partes: "`xx_YY.ZZZZ`".

valor locale	significado
`xx`	códigos de idioma ISO 639 (minúsculas) como em "en"
`YY`	códigos de idioma ISO 3166 (maiúsculas) como em "US"
`ZZZZ`	conjunto de codificação, definido sempre como "UTF-8"

Tabela 1.18: As 3 partes do valor locale

recomendação de locale	Idioma (área)
`en_US.UTF-8`	Inglês (EUA)
`en_GB.UTF-8`	Inglês (Grã-Bretanha)
`fr_FR.UTF-8`	Francês (França)
`de_DE.UTF-8`	Alemão (Alemanha)
`it_IT.UTF-8`	Italiano (Itália)
`es_ES.UTF-8`	Espanhol (Espanha)
`ca_ES.UTF-8`	Catalão (Espanha)
`sv_SE.UTF-8`	Sueco (Suécia)
`pt_BR.UTF-8`	Português (Brasil)
`ru_RU.UTF-8`	Russo (Rússia)
`zh_CN.UTF-8`	Chinês (Rep. Popular da China)
`zh_TW.UTF-8`	Chinês (Taiwan R.O.C.)
`ja_JP.UTF-8`	Japonês (Japão)
`ko_KR.UTF-8`	Coreano (República da Coreia)
`vi_VN.UTF-8`	Vietnamita (Vietname)

Tabela 1.19: Lista de recomendações de locale

A execução de comando típica utiliza uma sequência de linha de shell como o seguinte.

```
$ echo $LANG
en_US.UTF-8
$ date -u
Wed 19 May 2021 03:18:43 PM UTC
```

```
$ LANG=fr_FR.UTF-8 date -u
mer. 19 mai 2021 15:19:02 UTC
```

Aqui, o programa date(1) é executado com diferentes valores de configuração regional.

- Para o primeiro comando, "`$LANG`" é definida ao valor locale predefinido do sistema "`en_US.UTF-8`".
- Para o segundo comando, "`$LANG`" é definida ao valor locale UTF-8 Francês "`fr_FR.UTF-8`".

A maioria das execuções de comandos geralmente não têm definições de variáveis de ambiente precedentes. Para o exemplo acima, pode executar em alternativa o seguinte:

```
$ LANG=fr_FR.UTF-8
$ date -u
mer. 19 mai 2021 15:19:24 UTC
```

Dica
Quando preencher um relatório de bug, é uma boa ideia executar e verificar o comando em "`en_US.UTF-8`", se está num ambiente que não seja Inglês.

Para mais detalhes sobre configuração do locale, veja Secção 8.1.

1.5.3 A variável "$PATH"

Quando escreve um comando na shell, a shell procura o comando na lista de diretórios contida na variável de ambiente "`$PATH`". O valor da variável de ambiente "`$PATH`" também é chamado o caminho de procura da shell.

Na instalação Debian, por predefinição, a variável de ambiente "`$PATH`" das contas de utilizadores pode não incluir "`/usr/sbin`" nem "`/usr/sbin`". Por exemplo, o comando `ifconfig` necessita ser chamado com o caminho completo como "`/usr/sbin/ifconfig`". (De modo idêntico, o comando `ip` está localizado em "`/usr/bin`".)

Pode alterar a variável de ambiente "`$PATH`" da shell Bash pelos ficheiros "`~/.bash_profile`" ou "`~/.bashrc`".

1.5.4 A variável "$HOME"

Muitos comandos armazenam configurações específicas do utilizador no diretório home do utilizador e mudam o comportamento dele de acordo com o conteúdo dele. O diretório de utilizador é identificado pela variável de ambiente "`$HOME`".

valor de "$HOME"	situação de execução do programa
`/`	programa executado pelo processo de init (daemon)
`/root`	programa executado a partir da shell de root normal
`/home/normal_user`	programa executado a partir da shell de utilizador normal
`/home/normal_user`	programa executado a partir menu do ambiente GUI do utilizador normal
`/home/normal_user`	programa executado como root com o "`sudo program`"
`/root`	programa executado como root com o "`sudo -H program`"

Tabela 1.20: Lista de valores "`$HOME`"

Dica
A shell expande "`~/`" ao diretório home do utilizador atual, isto é, "`$HOME/`". A shell expande "`~foo/`" ao diretório home de `foo`, isto é, "`/home/foo/`".

Veja Secção 12.1.5 se `$HOME` não está disponível para o seu programa.

1.5.5 Opções da linha de comandos

Alguns comandos recebem argumentos. Os argumentos que começam com um "`-`" ou "`--`" são chamados opções e controlam o comportamento do comando.

```
$ date
Thu 20 May 2021 01:08:08 AM JST
$ date -R
Thu, 20 May 2021 01:08:12 +0900
```

Aqui o argumento de linha de comandos "`-R`" altera o comportamento de date(1) para gerar uma string da data compatível com RFC2822.

1.5.6 Glob da shell

Frequentemente deseja trabalhar com um conjunto de ficheiros sem os digitar a todos. O modelo de expansão do nome de ficheiro a utilizar a **glob** da shell , (por vezes referida como **wildcards**), facilita esta necessidade.

modelo glob da shell	**descrição de regra de correspondência**
*	nome de ficheiro não iniciado por "`.`"
`.*`	nome de ficheiro começado com "`.`" (em Bash 5.2 e posterior, "`.`" e "`..`" são excluídos por predefinição)
?	exatamente um caractere
`[…]`	exatamente um caractere com qualquer caractere envolvido em colchetes
`[a-z]`	exatamente um caractere com qualquer caractere entre "a" e "z"
`[!…]`	exatamente um caractere que não seja qualquer caractere envolvido em colchetes (excluíndo "`!`")

Tabela 1.21: Padrões glob da shell

Nota
Um "`/`" num nome de caminho não pode ser correspondido pelo padrão de glob da shell listado na tabela em cima.

Por exemplo, tente o seguinte:

```
$ mkdir foo; cd foo; touch b1.txt b2.txt b3.c b4.h .b5.txt ..b6.txt
$ mkdir bar; cd bar; touch b1.txt b2.txt b3.c b4.h .b5.txt ..b6.txt ; cd -
$ echo b*.txt
b1.txt b2.txt
$ echo b*
b1.txt b2.txt b3.c b4.h bar
$ echo */*.txt
bar/b1.txt bar/b2.txt
$ echo *.[hc]
b3.c b4.h
$ echo .*
.b5.txt ..b6.txt
$ echo b[!1-3]*
b4.h bar
$ echo b*/b[!1-3]*
bar/b4.h
$ cd ..; rm -rf junk
```

Veja glob(7).

Nota
Ao contrário da expansão de nome de ficheiro da shell, o modelo de shell "`*`" testado em find(1) com o teste "`-name`" etc., corresponde ao "`.`" inicial do nome de ficheiro. (Nova funcionalidade POSIX)

Nota
BASH pode ser moldado a alterar o comportamento de glob dele com as opções shopt dele embutidas como as "`dotglob`", "`noglob`", "`nocaseglob`", "`nullglob`", "`extglob`", etc. Veja bash(1).

1.5.7 Valor de retorno do comando

Cada comando retorna o estado de saída dele (na variável: "$?") como o valor de retorno.

estado de saída do comando	valor de retorno numérico	valor de retorno lógico
sucesso	zero, 0	**TRUE**
erro	não-zero, -1	**FALSE**

Tabela 1.22: Códigos de saída do comando

Por exemplo, tente o seguinte.

```
$ [ 1 = 1 ] ; echo $?
0
$ [ 1 = 2 ] ; echo $?
1
```

Nota
Por favor note que, no contexto lógico da shell, **sucesso** é tratado como o **VERDADEIRO** lógico o qual tem 0 (zero) como valor. De certa maneira isto não é intuitivo e necessita ser lembrado aqui.

1.5.8 Sequências de comandos típicas e redireccionamento da shell

Vamos tentar lembrar-nos dos seguintes idiomas de comando de shell escritos numa linha como parte de um comando de shell.

O sistema Debian é um sistema de multi-tarefa. Os trabalhos em segundo plano permitem aos utilizadores correrem vários programas numa única shell. A gestão dos processos em segundo plano envolve os embutidos da shell: `jobs`, `fg`, `bg` e `kill`. Por favor leia as secções de bash(1) sob "SINAIS", "CONTROLE DE TAREFAS" e builtins(1).

Por exemplo, tente o seguinte:

```
$ </etc/motd pager
```

```
$ pager </etc/motd
```

```
$ pager /etc/motd
```

```
$ cat /etc/motd | pager
```

idioma do comando	descrição
`command &`	execução **em segundo plano** do `command` na sub-shell
`command1 | command2`	liga em **pipe** a saída standard do `command1` à entrada standard do `command2` (execução **concorrente**)
`command1 2>&1 | command2`	liga em **pipe** a saídas standard e o erro standard do `command1` à entrada standard do `command2` (execução **concorrente**)
`command1 ; command2`	executa o `command1` e o `command2` **sequencialmente**
`command1 && command2`	executa o `command1`; se tiver sucesso, executa o `command2` **sequencialmente** (retorna sucesso se ambos `command1` **e** `command2` tiverem sucesso)
`command1 || command2`	executa o `command1`; se não tiver sucesso, executa o `command2` **sequencialmente** (retorna sucesso se o `command1` **ou** o `command2` tiverem sucesso
`command > foo`	redireciona a saída standard do `command` para o ficheiro `foo` (sobrescreve)
`command 2> foo`	redireciona o erro standard do `command` para o ficheiro `foo` (sobrescreve)
`command >> foo`	redireciona a saída standard do `command` para o ficheiro `foo` (acrescenta)
`command 2>> foo`	redireciona o erro standard do `command` para o ficheiro `foo` (acrescenta)
`command > foo 2>&1`	redireciona ambos saída standard e erro standard do `command` para o ficheiro `foo`
`command < foo`	redireciona a entrada standard do `command` ao ficheiro `foo`
`command << delimiter`	redireciona a entrada standard do `command` para as seguintes linhas até que o "`delimiter`" seja atingido (documentar aqui)
`command <<- delimiter`	redireciona a entrada standard do `command` às seguintes linhas até que o "`delimiter`" seja atingido (aqui documento, os caracteres tab de inicio são retirados das linhas de entrada)

Tabela 1.23: Idiomas de comandos de shell

Apesar dos 4 exemplos de redireccionamentos de shell mostrarem a mesma coisa, o último exemplo corre um comando `cat` extra e desperdiça recursos sem nenhuma razão.

A shell permite-lhe abrir ficheiros a usar o `exec` embutido com um descritor de ficheiro arbitrário.

```
$ echo Hello >foo
$ exec 3<foo 4>bar  # open files
$ cat <&3 >&4       # redirect stdin to 3, stdout to 4
$ exec 3<&- 4>&-    # close files
$ cat bar
Hello
```

O descritores de ficheiro 0-2 são predefinidos.

dispositivo	**descrição**	**descritor de ficheiro**
`stdin`	entrada standard	0
`stdout`	saída standard	1
`stderr`	erro standard	2

Tabela 1.24: Descritores de ficheiro predefinido

1.5.9 Comando alias

Pode definir um nome alternativo (alias) para um comando frequentemente utilizado.

Por exemplo, tente o seguinte:

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

Agora, "`la`" funciona como atalho para "`ls -la`" o que lista todos os ficheiros no formato de lista longa.

Pode listar quaisquer nomes alternativos existentes com o `alias` (veja bash(1) sob "COMANDOS EMBUTIDOS NA SHELL").

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

Pode identificar o caminho exacto ou a identidade do comando com `type` (veja bash(1) sob "COMANDOS EMBUTIDOS DA SHELL").

Por exemplo, tente o seguinte:

```
$ 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 o "`file`" não foi, assim o "`ls`" tem "hash", isto é, a shell tem um registo interno para o acesso rápido à localização do comando "`ls`".

Dica
Veja Secção 9.3.6.

1.6 Processamento de texto estilo Unix

Em ambientes de trabalho tipo Unix, o processamento de texto é feito ao canalizar texto por cadeias de ferramentas standard de processamento de texto. Esta foi outra inovação crucial do Unix.

1.6.1 Ferramentas de texto de Unix

Existem algumas ferramentas standard de processamento de texto que são muito usadas nos sistemas tipo Unix.

- Não é utilizada nenhuma expressão regular:
  - cat(1) concatena ficheiros e escreve o conteúdo inteiro.
  - tac(1) concatena ficheiros e escreve-os em reverso.
  - cut(1) seleciona partes de linhas e escreve-as.
  - head(1) escreve a parte inicial de ficheiros.
  - tail(1) escreve a parte final de ficheiros.
  - sort(1) organiza as linhas de ficheiros de texto.
  - uniq(1) remove linhas duplicadas de um ficheiro organizado.
  - tr(1) traduz ou apaga caracteres.
  - diff(1) compara ficheiros linha a linha.
- É utilizada uma expressão regular básica (**BRE**):
  - ed(1) é um editor de linhas primitivo.
  - sed(1) é um editor de streams.
  - grep(1) faz coincidir texto com padrões.
  - vim(1) é um editor de ecrã.
  - emacs(1) é um editor de écran. (**BRE** de certo modo extensa)
- É utilizada uma expressão regular extensa (**ERE**):
  - awk(1) faz processamento de texto simples.
  - egrep(1) faz coincidir texto com padrões.
  - `tcl`(3tcl) pode fazer todo o processamento de texto concebível: Veja re_syntax(3). Bastante usado com `tk`(3tk).
  - perl(1) pode fazer todo o processamento de texto concebível. Veja perlre(1).
  - pcre2grep(1) do pacote `pcre2-util` corresponde texto com padrões Perl Compatible Regular Expressions (PCRE).
  - python(1) com o módulo `re` pode fazer todo o processamento de texto concebível. Veja "`/usr/share/doc/python/h`

Se não tiver a certeza do que estes comandos fazem, por favor utilize "`man comando`" para descobri-lo por si.

Nota
A ordem de ordenação e a expressão de intervalo dependem da configuração regional. Se desejar obter um comportamento tradicional para um comando, utilize a configuração regional **C** ou **C.UTF-8** em vez da **UTF-8** normal (ver Secção 8.1).

Nota
As expressões regulares Perl (perlre(1)), Perl Compatible Regular Expressions (PCRE) e expressões regulares Python oferecidas pelo módulo `re` têm muitas extensões comuns ao **ERE** normal.

1.6.2 Expressões regulares

As expressões regulares são utilizadas em muitas ferramentas de processamento de texto. São análogas aos ”globs” da shell, mas são mais complicadas e poderosas.

A expressão regular descreve o modelo de correspondência e é feita de caracteres de texto e de **meta-caracteres**.

Um **meta-caractere** é apenas um caractere com um significado especial. Existem 2 estilos principais, **BRE** e **ERE**, a depender das ferramentas de texto conforme descrito acima.

BRE	ERE	descrição da expressão regular
`\ . [ ] ^ $ *`	`\ . [ ] ^ $ *`	**meta-caracteres** comuns
`\+ \? \( \) \{ \} |`		BRE apenas ”\” **metacaracteres** escapados
	`+ ? ( ) { } |`	ERE apenas ”\” **meta-caracteres** sem escape
c	c	corresponde a **não-meta-caractere** ”c”
`\c`	`\c`	corresponde a um caractere literal ”c” mesmo se ”c” é um **meta-caractere** por si só
`.`	`.`	corresponde a qualquer caractere incluindo nova linha
`^`	`^`	posição no início de uma cadeia
`$`	`$`	posição no fim de uma cadeia
`\<`	`\<`	posição no início de uma palavra
`\>`	`\>`	posição no final de uma palavra
`[abc…]`	`[abc…]`	corresponde a quaisquer caracteres em ”abc…”
`[^abc…]`	`[^abc…]`	corresponde a quaisquer caracteres excepto em ”abc…”
`r*`	`r*`	corresponde a zero ou mais expressões regulares identificadas por ”r”
`r\+`	`r+`	corresponde a uma ou mais expressões regulares identificadas por ”r”
`r\?`	`r?`	corresponde a zero ou uma expressão regular identificada por ”r”
`r1|r2`	`r1|r2`	corresponde a uma das expressões regulares identificadas por ”r1” ou ”r2”
`\(r1|r2\)`	`(r1|r2)`	corresponde a uma das expressões regulares identificadas por ”r1” ou ”r2” e trata-as como uma expressão regular **entre colchetes**

Tabela 1.25: Meta-caracteres para BRE e ERE

A expressão regular do **`emacs`** é basicamente **BRE** mas foi estendida para tratar ”+” e ”?” como **meta-caracteres** como em **ERE**. Assim, não há necessidade de os ’escapar’ com ”\” na expressão regular do `emacs`.

grep(1) pode ser utilizado para executar a pesquisa de texto com expressão regular.

Por exemplo, tente o seguinte:

```
$ 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
```

Dica
Veja Secção 9.3.6.

1.6.3 Expressões de substituição

Para a expressão de substituição, alguns caracteres têm significados especiais.

expressão de substituição	descrição do texto para substituir a expressão de substituição
`&`	que expressão regular corresponde (use `\&` no `emacs`)
`\n`	que nº **entre colchetes** da expressão regular correspondeu (a ser ”n” um número)

Tabela 1.26: A expressão de substituição

Para cadeia de substituição Perl, ”$&” é usado em vez de ”&” e ”$n” é usado em vez de ”\n”.

Por exemplo, tente o seguinte:

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

Aqui por favor preste atenção extra ao estilo da expressão regular **entre colchetes** e como as cadeias correspondentes são utilizadas no processo de substituição de texto nas diferentes ferramentas.

Estas expressões regulares também podem ser utilizadas para movimentos do cursor e acções de substituição de texto em alguns editores.

A barra descendente ”\” no fim da linha na linha de comandos da shell ’escapa’ a nova linha como um caractere de espaço em branco e continua a entrada na linha de comandos da shell na próxima linha.

Por favor leia todos os manuais relacionados para aprender estes comandos.

1.6.4 Substituição global com expressões regulares

O comando ed(1) pode substituir todas as instâncias de ”`FROM_REGEX`” por ”`TO_TEXT`” em ”`file`”.

```
$ ed file <<EOF
,s/FROM_REGEX/TO_TEXT/g
w
q
EOF
```

O comando sed(1) pode substituir todas as instâncias de "FROM_REGEX" por "TO_TEXT" em "file".

```
$ sed -i -e 's/FROM_REGEX/TO_TEXT/g' file
```

O comando vim(1) pode substituir todas as instâncias de "FROM_REGEX" com "TO_TEXT" em "ficheiro" ao usar comandos ex(1).

```
$ vim '+%s/FROM_REGEX/TO_TEXT/gc' '+update' '+q' file
```

Dica
A flag "c" em cima assegura confirmação interactiva para cada substituição.

Múltiplos ficheiros ("ficheiro1", "ficheiro2",e "ficheiro3") podem ser processados com expressões regulares à semelhança com vim(1) ou perl(1).

```
$ vim '+argdo %s/FROM_REGEX/TO_TEXT/gce|update' '+q' file1 file2 file3
```

Dica
A bandeira "e" em cima previne o erro "Nenhuma correspondência" de quebrar um mapeamento.

```
$ perl -i -p -e 's/FROM_REGEX/TO_TEXT/g;' file1 file2 file3
```

no exemplo perl(1), "-i" é para edição no-lugar de cada ficheiro objetivo e "-p" é para um ciclo implícito a todos os ficheiros fornecidos.

Dica
O uso do argumento "-i.bak" em vez de "-i" mantêm cada ficheiro original ao adicionar ".bak" ao seu nome de ficheiro. Isto torna a recuperação de erros mais fácil para substituições complexas.

Nota
ed(1) e vim(1) são **BRE**; perl(1) é **ERE**.

1.6.5 Extrair dados de tabela de ficheiro de texto

Vamos considerar um ficheiro de texto chamado "DPL" no qual alguns nomes de líderes de projectos Debian pré-2004 e as suas datas de iniciação estão listados num formato separado por espaços.

```
Ian     Murdock   August  1993
Bruce   Perens    April   1996
Ian     Jackson   January 1998
Wichert Akkerman  January 1999
Ben     Collins   April   2001
Bdale   Garbee    April   2002
Martin  Michlmayr March   2003
```

Dica
Veja "Uma História Breve de Debian" para o histórico de liderança de Debian mais recente.

O awk é frequentemente utilizado para extrair dados deste tipo de ficheiros.

Por exemplo, tente o seguinte:

```
$ awk '{ print $3 }' <DPL                  # month started
August
April
January
January
April
April
March
$ awk '($1=="Ian") { print }' <DPL         # DPL called Ian
Ian     Murdock   August  1993
Ian     Jackson   January 1998
$ awk '($2=="Perens") { print $3,$4 }' <DPL # When Perens started
April 1996
```

Shells como a Bash também podem ser utilizadas para analisar este tipo de ficheiro.

Por exemplo, tente o seguinte:

```
$ while read first last month year; do
    echo $month
  done <DPL
... same output as the first Awk example
```

Aqui, o comando embutido `read` usa caracteres em "`$IFS`" (separadores de campo internos) para dividir linhas em palavras.

Se alterar "`$IFS`" a "`:`", pode analisar "`/etc/passwd`" facilmente com a shell.

```
$ oldIFS="$IFS"   # save old value
$ IFS=':'
$ while read user password uid gid rest_of_line; do
    if [ "$user" = "bozo" ]; then
      echo "$user's ID is $uid"
    fi
  done < /etc/passwd
bozo's ID is 1000
$ IFS="$oldIFS"   # restore old value
```

(Se o Awk for utilizado para fazer o equivalente, utilize "`FS=':'`" para definir o campo separador.)

O IFS também é usado pela shell para dividir resultados de expansão de parâmetros, substituição de comandos e expansão aritmética. Estas não ocorrem em palavras dentro de citações simples ou duplas. O valor predefinido do IFS é *espaço*, *tab* e *nova-linha* combinados.

Tenha cuidado ao usar estes truques IFS da shell. Podem acontecer coisas estranhas, quando a shell interpreta partes do script como a **entrada** dela.

```
$ IFS=":,"                        # use ":" and "," as IFS
$ echo IFS=$IFS,   IFS="$IFS"     # echo is a Bash builtin
IFS=  , IFS=:,
$ date -R                         # just a command output
Sat, 23 Aug 2003 08:30:15 +0200
$ echo $(date -R)                 # sub shell --> input to main shell
Sat  23 Aug 2003 08 30 36 +0200
$ unset IFS                       # reset IFS to the default
$ echo $(date -R)
Sat, 23 Aug 2003 08:30:50 +0200
```

1.6.6 Trechos de script para canalizar comandos em pipe

Os seguintes scripts fazem coisas bonitas como parte de um pipe.

trecho de script (escrito numa linha)	**efeito do comando**
`find /usr -print`	encontra todos os ficheiros sob "`/usr`"
`seq 1 100`	escreve 1 até 100
`| xargs -n 1 command`	corre o comando repetidamente com cada item do pipe como seu argumento
`| xargs -n 1 echo`	divide itens separados por espaços do pipe em linhas
`| xargs echo`	junta todas as linhas do pipe numa linha
`| grep -e regex_pattern`	extrai as linhas do pipe que contêm o *padrão_da_expressão_regular*
`| grep -v -e regex_pattern`	extrai as linhas do pipe que não contêm o *padrão_da_expressão_regular*
`| cut -d: -f3 -`	extrai do pipe o terceiro campo separado por "`:`" (ficheiro passwd etc.)
`| awk '{ print $3 }'`	extrai do pipe o terceiro campo separado por espaços
`| awk -F'\t' '{ print $3 }'`	extrai do pipe o terceiro campo separado por tab
`| col -bx`	remove os backspace e expande as tabs para espaços
`| expand -`	expande separadores
`| sort| uniq`	organiza e remove duplicados
`| tr 'A-Z' 'a-z'`	converte maiúsculas para minúsculas
`| tr -d '\n'`	concatena linhas numa linha
`| tr -d '\r'`	remover CR
`| sed 's/^/# /'`	adiciona "#" ao inicio de cada linha
`| sed 's/\.ext//g'`	remover "`.ext`"
`| sed -n -e 2p`	escreve a segunda linha
`| head -n 2 -`	escreve as primeiras duas linhas
`| tail -n 2 -`	escreve as últimas duas linhas

Tabela 1.27: Lista de trechos de script para canalizar comandos em pipe

Um script de shell de uma linha pode fazer ciclos sobre muitos ficheiros a usar o find(1) e xargs(1) para executar tarefas bastante complicadas. Veja Secção 10.1.5 e Secção 9.4.9.

Quando a utilização dos modos interativos da shell se torna muito complicada, por favor considere escrever um script de shell (veja Secção 12.1).

Capítulo 2

Gestão de pacotes Debian

Nota
Este capítulo é escrito a assumir que o lançamento estável mais recente tem o nome de código: `trixie`.
A fonte de dados do sistema APT é coletivamente referida como **a lista de fontes** neste documento. Isto pode ser definido em qualquer parte do ficheiro "`/etc/apt/sources.list`", "`/etc/apt/sources.list.d/*.list`" ou "`/etc/apt/sources.list.d/*.sources`".

2.1 Pré-requisitos da gestão de pacotes Debian

2.1.1 Sistema de gestão de pacotes Debian

Debian é uma organização voluntária que constrói distribuições **consistentes** de pacotes binários pré-compilados de software livre e distribui-os a partir do arquivo dele.

O arquivo Debian é oferecido por muitos sites mirror remotos para acesso através de métodos HTTP e FTP. Também está disponível em CD-ROM/DVD.

O atual sistema de gestão de pacotes Debian que pode utilizar todos estes recursos é o Ferramenta de instalações avançada (APT).

O sistema de gestão de pacotes Debian, **quando utilizado de modo apropriado**, oferece ao utilizador o instalar de **conjuntos consistentes de pacotes binários** no sistema a partir do arquivo. atualmente, existem 78072 pacotes disponíveis para a arquitectura amd64.

O sistema de gestão de pacotes Debian tem um histórico rico e muitas opções de escolha para o programa cliente do utilizador e de método de acesso ao arquivo. Atualmente, recomendamos o seguinte:

- apt(8) para todas as operações de linha de comandos interativas, incluindo instalação, remoção e atualização de pacotes.
- apt-get(8) para chamar o sistema de gestão de pacotes Debian a partir de scripts. É também uma opção regressiva quando o `apt` não está disponível (comum em sistemas Debian antigos).
- aptitude(8) para uma interface de texto interactiva para gerir os pacotes instalados e procurar os pacotes disponíveis.

pacote	popcon	tamanho	descrição
dpkg	V:906, I:1000	4944	sistema de gestão de pacotes de baixo nível para Debian (baseado em ficheiros)
apt	V:895, I:1000	4956	Frontend do APT para gerir pacotes com CLI: apt/apt-get/apt-cache
aptitude	V:31, I:160	4621	Interface do APT para gerir pacotes interativamente com uma consola de ecrã completo: aptitude(8)
tasksel	V:38, I:985	351	Front-end APT para instalar tarefas selecionadas: tasksel(8)
unattended-upgrades	V:130, I:182	320	pacote de melhoria para o APT para ativar a instalação automática de atualizações de segurança
gnome-software	V:158, I:263	4732	Centro de Software para GNOME (GUI APT front-end)
synaptic	V:31, I:269	7264	gestor de pacotes gráfico (interface GTK para APT)
apt-utils	V:420, I:998	1181	Programas utilitários do APT: apt-extracttemplates(1), apt-ftparchive(1) e apt-sortpkgs(1)
apt-listchanges	V:403, I:894	562	ferramenta de notificação do histórico de alterações do pacote
apt-listbugs	V:4.7, I:6.5	512	lista bugs críticos antes de cada instalação do APT
apt-file	V:14, I:53	76	Utilitário de busca de pacotes do APT - interface de linha de comandos
apt-rdepends	V:0.4, I:4.2	39	lista recursivamente dependências de pacotes

Tabela 2.1: Lista de ferramentas de gestão de pacotes Debian

2.1.2 Configuração de pacotes

Aqui estão alguns pontos chave para a configuração de pacotes no sistema Debian.

- Para um Ambiente de Trabalho moderno, reiniciar o sistema após uma alteração de configuração de pacote e atualização de pacote é uma boa ideia ara assegurar que o sistema vai funcionar de modo apropriado.
- A configuração manual feita pelo administrador do sistema é respeitada. Por outras palavras, o sistema de configuração de pacotes não faz configurações intrusivas por conveniência.
- Cada pacote vem com o próprio script de configuração com a interface de utilizador standard chamada debconf(7) para ajudar no processo inicial de instalação do pacote.
- Os Programadores do Debian dão o melhor para tornar a sua experiência de atualização isenta de falhas com scripts de configuração de pacotes.
- As funcionalidades totais do software empacotado estão disponíveis ao administrador do sistema. Mas aquelas com riscos de segurança estão desactivadas na instalação predefinida.
- Se manualmente ativou um serviço com alguns riscos de segurança você é o responsável pelo confinamento de risco.
- A configuração esotérica pode ser activada manualmente pelo administrador do sistema. Isto pode criar interferência com programas de ajuda genéricos populares para a configuração do sistema.

2.1.3 Precauções básicas



Atenção
Não instale pacotes de misturas aleatórias de suites. Provavelmente vai quebrar a consistência do pacote o que requer conhecimentos profundos de gestão do sistema tais como ABI compilador, versão de biblioteca, funcionalidades do interpretador, etc.

O administrador de sistemas Debian novato deve ficar com a suite **`stable`** de Debian enquanto aplica apenas atualizações de segurança. Até compreender muito bem o sistema Debian, você deve seguir as seguintes precauções.

- Não inclua **`testing`** ou **`unstable`** na **lista de fontes**.
- Não misture o Debian padrão com outros repositórios não-Debian como o Ubuntu na **lista de fontes**.
- Não crie "`/etc/apt/preferences`".
- Não altere o comportamento predefinido das ferramentas de gestão de pacotes através de ficheiros de configuração sem conhecer os seus impactos totais.
- Não instale pacotes aleatórios com "`dpkg -i` *`qualquer_pacote`*".
- Nunca instale pacotes aleatórios com "`dpkg --force-all -i` *`qualquer_pacote`*".
- Não apague ou altere os ficheiros em "`/var/lib/dpkg/`".
- Não sobrescreva ficheiros do sistema ao instalar programas compilados a partir do código-fonte.
  - Instale-os em "`/usr/local`" ou "`/opt`", se necessário.

Os efeitos não-compatíveis causados pela violação das precauções acima para com o sistema de gestão de pacotes Debian podem deixar o seu sistema inutilizável.

O administrador de sistema Debian sério que corre servidores de missões críticas, deve tomar precauções extra.

- Não instale nenhuns pacotes incluindo as atualizações de segurança da Debian sem os testar completamente com as suas configurações particulares sob condições seguras.
  - Como o administrador do sistema é o responsável final pelo seu sistema.
  - A longa história de estabilidade do sistema Debian não é uma garantia por si só.

2.1.4 A vida com atualizações eternas



Cuidado
Para o seu **servidor de produção** a suite `stable` é recomendada com as atualizações de segurança. O mesmo pode ser dito para PCs de ambiente de trabalho onde pode dispor de esforços limitados de administração.

Apesar dos meus avisos acima, sei que muitos leitores deste documento podem querer executar as suites `testing` ou `unstable` mais recentes.

O Esclarecimento com o seguinte salva uma pessoa do karma da eterna luta do inferno das atualizações e permite-lhe alcançar o nirvana de Debian.

Esta lista destina-se ao ambiente de trabalho **auto-administrado**.

- Use a suite `testing` já que esta é na prática a atualização contínua gerida automaticamente pela infraestrutura de controlo de qualidade do repositório Debian, como a integração contínua Debian, as práticas de upload somente de código fonte e o acompanhamento de transição de bibliotecas. Os pacotes na suite `testing` são atualizados com frequência suficiente para oferecer todos os recursos mais recentes.
- Definir o nome de código correspondente ao conjunto `testing` ("`forky`" durante o ciclo de lançamento `trixie-as-stable`) na **lista de fontes**.
- Atualize manualmente este nome de código na **lista de fontes** para o novo apenas após avaliar a situação por si mesmo por cerca de um mês após o lançamento da suíte principal. A lista de discussão de utilizadores e desenvolvedores Debian são boas fontes de informação para isso também.

A utilização da suite `unstable` não é recomendada. A suite `unstable` (instável) é **boa para depurar pacotes** como programador mas tende a expor a riscos desnecessários para a utilização normal do ambiente de trabalho. Mesmo que a suite `unstable` do sistema Debian pareça muito estável na maioria das vezes, houve alguns problemas com pacotes e alguns deles não foram tão triviais de resolver.

Aqui estão algumas idéias básicas de medidas de precaução para assegurar uma recuperação rápida e fácil de bugs em pacotes Debian.

- Torne o sistema de salvamento disponível seguindo Secção 3.2.
- Faça um sistema de **duplo arranque** ao instalar a suite `stable` do sistema Debian noutra partição
- Considere instalar o `apt-listbugs` para verificar informação do Debian Bug Tracking System (BTS) antes das atualizações
- Conheça o suficiente da infraestrutura do sistema de pacotes para contornar o problema



Cuidado
Se não conseguir fazer nenhumas destas ações de precaução, provavelmente não está preparado para as suites `testing` e `unstable`.

2.1.5 Básico do arquivos Debian

Dica
A política oficial do arquivo Debian está definida em Manual de Políticas Debian, Capítulo 2 - O Arquivo Debian.

Vamos olhar ao arquivo Debian a partir da perspectiva do utilizador do sistema.

Para um utilizador do sistema, o arquivo Debian é acedido utilizando o sistema APT.

O sistema APT especifica a sua fonte de dados como a **lista de fontes** e esta é descrita em sources.list(5).

Para o sistema `trixie` com o acesso HTTP típico, **a lista de fontes** é fornecida no estilo deb822 moderno em "`/etc/apt/sources.list.d/debian.sources`" como o seguinte:

```
Types: deb deb-src
URIs: http://deb.debian.org/debian/
Suites: trixie trixie-updates
Components: main non-free-firmware contrib non-free

Types: deb deb-src
URIs: http://security.debian.org/debian-security/
Suites: trixie-security
Components: main non-free-firmware contrib non-free
```

Dica
Se a **lista de fontes** for fornecida no antigo estilo descontinuado de uma linha em ficheiros "`/etc/apt/sources.list`" ou "`/etc/apt/sources.list.d/*.list`", atualize-os com:

```
$ sudo apt modernize-sources
```

Os pontos principais da **lista de fontes** em estilo deb822 são os seguintes.

- Os seus ficheiros de definição estão nos ficheiros "`/etc/apt/sources.list.d/*.sources`".
- Cada bloco de linhas separado por uma linha em branco define a fonte de dados para o sistema APT.
- A estrofe "`Types:`" define a lista de tipos tais como "`deb`" e "`deb-src`".
- A estrofe "`URIs:`" define a lista de URIs raiz do repositório Debian.
- A estrofe "`Suites:`" define a lista de nomes de distribuição utilizando o nome da suite ou o nome de código.
- A estrofe "`Components:`" define a lista de nomes de áreas de arquivo válidas do arquivo Debian.

A definição para "`deb-src`" pode ser omitida com segurança se for apenas para `aptitude` que não acede aos meta dados relacionados com a fonte. Acelera as atualizações dos meta dados do arquivo.

O URL pode ser "`https://`", "`http://`", "`ftp://`", "`file://`", ...

As linhas que começam por "#" são comentários e são ignoradas.

Aqui, tenho tendência para utilizar o nome de código "`trixie`" ou "`forky`" em vez do nome da suite "`stable`" ou "`testing`" para evitar surpresas quando a próxima `stable` for lançada.

Dica
Se "`sid`" for utilizado no exemplo acima em vez de "`trixie`", a linha "`deb: http://security.debian.org/ ...`" ou o seu conteúdo equivalente em deb822 para atualizações de segurança na **lista de fontes** não é necessária. Isto porque não existe nenhum arquivo de atualizações de segurança para "`sid`" (`unstable`).

Aqui está a lista de URLs dos sites de arquivo Debian e o nome da suite ou nome de código usado no ficheiro de configuração após o lançamento do `trixie`.

URL do arquivo	nome da suite	codinome	objetivo do repositório
http://deb.debian.org/debian/	`stable`	`trixie`	Quase-estática `stable` realizada após controlos exaustivos
http://deb.debian.org/debian/	`stable-updates`	`trixie-updates`	Atualizações urgentes para a suite `stable` (importante)
http://security.debian.org/debian-security/	`stable-security`	`trixie-security`	Atualizações de segurança para a suite `stable` (importante)
http://deb.debian.org/debian/	`testing`	`forky`	Disponibilização dinâmica da suite `testing` após controlos decentes e prazos curtos
http://deb.debian.org/debian/	`unstable`	`sid`	Lançamento `unstable` dinâmico após verificações mínimas e sem esperas
http://deb.debian.org/debian/	`experimental`	N/D	Experiências de pré-lançamento efetuadas por programadores (opcional, apenas para programadores)
http://deb.debian.org/debian/	`stable-proposed-updates`	`trixie-proposed-updates`	Atualizações propostas para a próxima versão `stable` (opcional)
http://deb.debian.org/debian/	`stable-backports`	`trixie-backports`	Coleção aleatória de pacotes recompilados maioritariamente da versão `testing` (opcional)
http://security.debian.org/debian-security/	`testing-security`	`forky-security`	Não é ativamente apoiado nem utilizado pela equipa de segurança

Tabela 2.2: Lista de sites de arquivos Debian



Cuidado
Apenas a suite **`stable`** pura com atualizações de segurança proporciona a melhor estabilidade. Executar maioritariamente a suite **`stable`** misturada com alguns pacotes da **`testing`** ou da suite **`unstable`** é mais arriscado do que executar a versão **`unstable`** pura devido a incompatibilidade de versões de bibliotecas, etc. Se precisa mesmo da última versão de alguns programas sob lançamento **`stable`**, por favor use pacotes dos serviços stable-updates e backports (veja Secção 2.7.4). Estes serviços devem ser usados com cuidado extra.



Cuidado
Basicamente deve listar apenas uma das suites `stable`, `testing`, ou `unstable` na linha "`deb`". Se listar qualquer combinação das suites `stable`, `testing` e `unstable` na linha "`deb`", os programas do APT abrandam enquanto apenas o arquivo mais recente é efectivo. Faz sentido várias listagens quando o ficheiro "`/etc/apt/preferences`" é utilizado com objetivos claros (veja Secção 2.7.7).

Dica
Para o sistema Debian com a suite `stable`, é uma boa ideia incluir o conteúdo com "`http://security.debian.org/`" na **lista de fontes** para ativar as atualizações de segurança como no exemplo acima.

Nota
Os bugs de segurança para a suite `stable` são corrigidos pela equipa de segurança Debian. Esta atividade tem sido bastante rigorosa e fiável. Os existentes na suite `testing` podem ser corrigidos pela equipa de segurança de Debian testing. Por várias razões, esta atividade não é tão rigorosa como a da `stable` e poderá ter de esperar pela migração dos pacotes de `unstable` corrigidos para o arquivo `testing`. Os existentes no repositório `unstable` são corrigidos pelo responsável individual. Pacotes `unstable` mantidos ativamente estão normalmente em boa forma, aproveitando as últimas correções de segurança do autor. Veja a FAQ de segurança do Debian para saber como o Debian lida com bugs de segurança.

área	quantidade de pacotes	critério do componente do pacote
`main`	76587	em conformidade com DFSG e nenhuma dependência a `non-free`
`non-free-firmware`	78	não compatível com DFSG, firmware necessário para uma experiência razoável de instalação do sistema
`contrib`	362	em conformidade com DFSG mas com dependências a `non-free`
`non-free`	1045	não compatível com DFSG e não em `non-free-firmware`

Tabela 2.3: Lista de área de arquivo Debian

Aqui a quantidade de pacotes em cima é para a arquitectura amd64. A área `main` disponibiliza o sistema Debian (veja Secção 2.1.6).

A organização do arquivo Debian pode ser melhor estudada ao apontar o seu navegador a cada URL de arquivo seguido de `dists` ou `pool`.

A distribuição é referida de duas maneiras, a suite ou o nome-de-código. A palavra distribuição é usada alternativamente como o sinónimo de suite em muitas documentações. A relação entre a suite e o nome de código pode ser resumida ao seguinte.

A história dos nomes de código está descrita em Debian FAQ: 6.2.1 Que outros nomes de código foram usados no passado?

Tempo	suite = `stable`	suite = `testing`	suite = `unstable`
após o lançamento `trixie`	nome de código = `trixie`	nome de código = `forky`	nome de código = `sid`
após o lançamento `forky`	nome de código = `forky`	nome de código = `duke`	nome de código = `sid`

Tabela 2.4: A relação entre suite e nome de código

Na terminologia estrita do arquivo Debian, a palavra ”secção” é utilizada especialmente para categorizar os pacotes pela área de aplicação. (Apesar da palavra ”secção main” poder por vezes ser utilizada para descrever a área do arquivo Debian com o nome ”main”.)

Cada vez que é feito um novo upload por um programador de Debian (DD) para o arquivo `unstable` (por processamento do incoming), é necessário que o DD assegure que os pacotes enviados sejam compatíveis com o conjunto de pacotes mais recente no arquivo `unstable` mais recente.

Se o DD quebrar esta compatibilidade intencionalmente para uma atualização importante de biblioteca ou etc., geralmente existe um anúncio na lista de email debian-devel etc.

Antes que um conjunto de pacotes seja movido pelo script de manutenção do arquivo Debian do arquivo `unstable` para o arquivo `testing`, o script de manutenção do arquivo não verifica apenas a maturidade (cerca de 2-10 dias de idade) e o estado dos relatórios de bug RC para os pacotes mas também tenta assegurar que sejam compatíveis com o conjunto de pacotes mais recente no arquivo `testing`. Este processo torna o arquivo `testing` muito atual e utilizável.

Através do processo de congelamento gradual do arquivo liderado pela equipa de lançamento, o arquivo `testing` é amadurecido para o tornar completamente consistente e livre de bugs com algumas intervenções manuais. Então o novo lançamento `stable` é criado ao atribuir o nome de código do antigo arquivo `testing` ao novo arquivo `stable` e a criar um novo nome de código para o novo arquivo `testing`. O conteúdo inicial do novo arquivo `testing` é exatamente o mesmo que o arquivo `stable` recentemente lançado.

Ambos os arquivos `unstable` e `testing` podem sofrer falhas temporárias devido a vários fatores:

- Envio de pacotes danificados ao arquivo (maioritariamente para `unstable`)
- Atraso de aceitação dos novos pacotes no arquivo (maioritariamente para `unstable`)
- Problemas com o tempo de sincronização do arquivo (tanto para `testing` como `unstable`)
- Intervenção manual no arquivo, tal como remoção de pacotes (mais para `testing`) etc.

Se alguma vez decidir utilizar estes arquivos, deverá ser capaz de corrigir ou contornar este tipo de problemas.



Cuidado
Durante alguns meses após um novo lançamento de `stable`, a maioria dos utilizadores de ambientes de trabalho devem usar o arquivo `stable` com as atualizações de segurança dele mesmo que normalmente usem os arquivos `unstable` ou `testing`. Durante este período de transição, ambos arquivos `unstable` e `testing` não são bons para a maioria das pessoas. O seu sistema é difícil de manter em boas condições de funcionamento com o arquivo `unstable` porque sofre de vagas de grandes atualizações nos pacotes principais. O arquivo `testing` também não é útil porque contém praticamente o mesmo conteúdo que o arquivo `stable` mas sem o suporte de segurança dele (Anúncio-de-segurança-de-testing-Debian-2008-12). Após um mês ou mais, os arquivos de `unstable` ou `testing` podem ser úteis se for cuidadoso.

Dica
Quando se acompanha o arquivo `testing`, um problema causado por um pacote removido é geralmente contornado ao instalar o pacote correspondente do arquivo `unstable` que foi lançado para correcção de bug.

Veja Manual de Políticas Debian para as definições do arquivo.

- "Secções"
- "Prioridades"
- "Sistema base"
- "Pacotes essenciais"

2.1.6 Debian é 100% software livre

Debian é 100% software livre por causa do seguinte:

- Por predefinição, Debian instala apenas software livre para respeitar as liberdades dos utilizadores.
- Debian disponibiliza apenas software livre no `main`.
- Debian recomenda correr apenas software livre do `main`.
- Nenhum pacote `main` depende nem recomenda pacotes na `non-free` nem na `non-free-firmware` nem na `contrib`.

Algumas pessoas pensam se os 2 seguintes factos se contradizem ou não.

- "Debian irá manter-se 100% livre". (Primeiro termo do Debian Social Contract)
- Os servidores Debian hospedam alguns pacotes `non-free-firmware`, `non-free` e `contrib`.

Estes não se contradizem, devido ao seguinte.

- O sistema Debian é 100% livre e os seus pacotes estão alojados em servidores Debian na área `main`.
- Pacotes fora do sistema Debian são hospedados por servidores Debian nas áreas `non-free`, `non-free-firmware` e `contrib`.

Isto é perfeitamente explicado nos termos 4º e 5º do Debian Social Contract:

- As nossas prioridades são os nossos utilizadores e o software livre
  - Seremos guiados pelas necessidades dos nossos utilizadores e da comunidade de software livre. Iremos pôr o interesse deles no topo das nossas prioridades. Iremos suportar as necessidades dos nossos utilizadores para operação em muitos ambientes de computação distintos. Não nos oporemos a software não-livre que se destine a ser utilizado em sistemas Debian, nem tentaremos cobrar qualquer taxa a pessoas que criem ou utilizem tais trabalhos. Iremos permitir que terceiros criem distribuições a conter o sistema Debian com outros trabalhos, sem qualquer taxa para nós. Para apoio destes objetivos, iremos disponibilizar um sistema integrado de materiais de alta qualidade sem restrições legais que previnam tais utilizações do sistema.
- Trabalhos que não coincidem com os nossos standards de software livre
  - Nós reconhecemos que alguns de nossos utilizadores requerem o uso de trabalhos que não estão de acordo com a Definição Debian de Software Livre. Nós criamos áreas "`non-free`", "`non-free-firmware`" e "`contrib`" no nosso repositório para estes trabalhos. Os pacotes nestas áreas não são parte do sistema Debian, embora tenham sido configurados para uso com o Debian. Nós encorajamos os fabricantes de CDs a ler as licenças dos pacotes nestas áreas e determinar se eles podem distribuir os pacotes em seus CDs. Assim, embora os trabalhos não-livres não façam parte da Debian, nós suportamos a sua utilização e fornecemos infraestruturas para pacotes não-livres (tais como o nosso sistema de acompanhamento de bugs e listas de correio eletrónico). A média oficial da Debian pode incluir firmware que de outra forma não é parte do sistema Debian para permitir o uso da Debian com hardware que requer tal firmware.

Nota
O texto atual do quinto termo no atual Contrato Social Debian 1.2 é ligeiramente diferente do texto acima. Este desvio editorial é intencional para tornar este documento do utilizador consistente sem alterar o conteúdo real do Contrato Social.

Os utilizadores devem estar cientes dos riscos de usar pacotes das áreas `non-free`, `non-free-firmware` e `contrib`:

- falta de liberdade para tais pacotes de software
- falta de suporte Debian em tais pacotes de software (Debian não pode suportar devidamente software sem ter acesso ao seu código-fonte.)
- contaminação do seu sistema Debian 100% livre

As Debian Free Software Guidelines são os standards de software livre para Debian. Debian interpreta ”software” no âmbito mais amplo incluindo documentação, firmware, logo e dados artísticos no pacote. Isto torna os standards de software livre de Debian muito rigorosos.

Tipicamente os pacotes da `non-free`, `non-free-firmware` e da `contrib` incluem pacotes distribuídos livremente dos seguintes tipos:

- Pacotes de documentação sob a GNU Free Documentation License com secções invariantes tais como as do GCC e do Make. (a maioria encontra-se na secção `non-free/doc`.)
- Pacotes de firmware contendo dados binários sem fonte, tais como os listados em Secção 9.10.5 como `non-free-firmw` (maioritariamente encontrados na secção `non-free-firmware/kernel`.)
- Pacotes de fontes e jogos com restrições comerciais de utilização e/ou modificação de conteúdo.

Por favor note que a quantidade de pacotes de `non-free`, `non-free-firmware` e `contrib` é menos de 2% dos pacotes de `main`. Ativar o acesso às áreas `non-free`, `non-free-firmware` e `contrib` não turva a fonte dos pacotes. A utilização do ecrã interativo do aptitude(8) disponibiliza-lhe visibilidade e controlo total sobre que pacotes estão instalados e a partir de qual das áreas, para manter o seu sistema livre conforme desejar.

2.1.7 Dependências de pacote

O sistema Debian oferece um conjunto consistente de pacotes binários através do mecanismo de declaração de dependências binárias dele com versões nos campos do ficheiro de controle. Aqui está uma definição deles um pouco simplificada:

- ”Dependências”
  - Isto declara uma dependência absoluta e todos os pacotes listados neste campo têm de ser instalados ao mesmo tempo ou com antecedência.
- ”Pré-Dependências”
  - Isto é como o Depends, excepto que requer a instalação completa de todos os pacotes listados com antecedência.
- ”Recomendados”
  - Isto declara uma dependência forte mas não absoluta. A maioria dos utilizadores não iriam querer o pacote a menos que todos os pacotes listados neste campo estejam instalados.
- ”Sugeridos”

  - Isto declara uma dependência fraca. Muitos utilizadores deste pacote podem beneficiar ao instalar os pacotes listados neste campo mas podem ter as funções razoáveis sem eles.
- ”Melhoramentos”
  - Isto declara uma dependência fraca como Sugerida mas funciona na direção oposta.
- ”Quebras”
  - Isto declara uma incompatibilidade do pacote normalmente com alguma especificação de versão. Geralmente a resolução é atualizar todos os pacotes listados neste campo.
- ”Conflitos”
  - Isto declara uma incompatibilidade absoluta. Todos os pacotes listados neste campo têm de ser removidos para instalar este pacote.
- ”Substitui”
  - Isto é declarado quando os ficheiros instalados por este pacote substituem ficheiros nos pacotes listados.
- ”Proporciona”
  - Isto é declarado quando este pacote disponibiliza todos os ficheiros e funcionalidades dos pacotes listados.

Nota
Por favor note que definir ”Provides”, ”Conflicts” e ”Replaces” em simultâneo a um pacote virtual é a configuração sã. Isto assegura que apenas um pacote real que disponibilize este pacote virtual possa ser instalado de cada vez.

A definição oficial incluindo dependências de fonte encontra-se em O Manual de Políticas: Capítulo 7 - Declarar relações entre pacotes.

2.1.8 O fluxo de eventos da gestão de pacotes

Aqui está um resumo do fluxo de eventos simplificado da gestão de pacotes pelo APT.

- **Update** (”`apt update`”, ”`aptitude update`” ou ”`apt-get update`”):
  1. Obtém meta-dados do arquivo a partir do arquivo remoto
  2. Re-constrói e atualiza os meta-dados locais para utilização do APT
- **Upgrade** (”`apt upgrade`” e ”`apt full-upgrade`”, ou ”`aptitude safe-upgrade`” e ”`aptitude full-upgrade`”, ou ”`apt-get upgrade`” e ”`apt-get dist-upgrade`”):
  1. Escolhe a versão candidata que geralmente é a versão mais recente disponível para todos os pacotes instalados (veja Secção 2.7.7 para exceções)
  2. Resolve a dependência do pacote
  3. Obtém os pacotes binários selecionados do arquivo remoto se a versão candidata diferir da versão instalada
  4. Desempacota os pacotes binários obtidos
  5. Corre o script **preinst**
  6. Instala os ficheiros binários
  7. Corre o script **postinst**
- **Instalar** (”`apt install …`”, `aptitude install …`” ou ”`apt-get install …`”):
  1. Escolhe os pacotes listados na linha de comandos

2. Resolve a dependência do pacote
3. Obtém os pacotes binários selecionados a partir do arquivo remoto
4. Desempacota os pacotes binários obtidos
5. Corre o script **preinst**
6. Instala os ficheiros binários
7. Corre o script **postinst**

- **Remover** ("`apt remove` …", "`aptitude remove` …" ou "`apt-get remove` …"):
  1. Escolhe os pacotes listados na linha de comandos
  2. Resolve a dependência do pacote
  3. Corre o script **prerm**
  4. Remove os ficheiros instalados **excepto** os ficheiros de configuração
  5. Corre o script **postrm**
- **Purgar** ("`apt purge`", "`aptitude purge` …" ou "`apt-get purge` …"):
  1. Escolhe os pacotes listados na linha de comandos
  2. Resolve a dependência do pacote
  3. Corre o script **prerm**
  4. Remove os ficheiros instalados **incluindo** os ficheiros de configuração
  5. Corre o script **postrm**

Aqui, saltei intencionalmente detalhes técnicos por causa da visão geral.

2.1.9 Primeira resposta a problemas com a gestão de pacotes

Deve ler a boa documentação oficial. O primeiro documento a ler é específico de Debian "`/usr/share/doc/`*`package_nam`*
Também deve ser consultada outra documentação em "`/usr/share/doc/`*`package_name/`*". Se definir a shell como Secção 1.4.2, escreva o seguinte.

```
$ cd package_name
$ pager README.Debian
$ mc
```

Pode necessitar instalar o pacote de documentação correspondente, com o sufixo "`-doc`" no nome, para informações mais detalhadas.

Se estiver a ter problemas com um pacote específico, certifique-se que verifica primeiro o Sistema de acompanhamento de bugs Debian (BTS).

site web	**comando**
Página inicial do Sistema de acompanhamento de bugs Debian (BTS)	`sensible-browser "https://bugs.debian.org/"`
O relatório de bug de um nome de pacote conhecido	`sensible-browser "https://bugs.debian.org/`*`package_name`*`"`
O relatório de bug de uma quantidade de bugs conhecida	`sensible-browser "https://bugs.debian.org/`*`bug_number`*`"`

Tabela 2.5: Lista de sites web chave para resolver problemas com um pacote específico

Procure no Google com palavras de busca incluindo "`site:debian.org`", "`site:wiki.debian.org`", "`site:lists.de`
etc.

Quando criar um relatório de bug, por favor use o comando reportbug(1).

2.1.10 Como escolher os pacotes Debian

Quando encontrar mais de 2 pacotes semelhantes e não sabe qual deles instalar sem o esforço de ”teste e erro”, deve utilizar algum **senso comum**. Considero os seguintes pontos como boas indicações dos pacotes preferidos:

- Essencial: sim > não
- Área: main > contrib > non-free
- Prioridade: required > important > standard > optional > extra
- Tasks: pacotes listados em tarefas como ”Ambiente de Trabalho”
- Pacotes selecionados pelo pacote de dependência (ex., `gcc-16` por `gcc`)
- Popcon: mais alto na votação e número de instalações
- Changelog: atualizações regulares feitas pelo responsável do pacote
- BTS: Nenhum bug RC (nenhum crítico, nenhum grave e nenhum bug sério)
- BTS: manutenção responsável dos relatórios de bugs
- BTS: maior quantidade de bugs corrigidos recentemente
- BTS: menor quantidade de bugs ”não-lista-de-desejos” remanescentes

O Debian, que é um projecto voluntário com modelo de desenvolvimento distribuído, o arquivo dele contém muitos pacotes com diferentes objetivos e qualidade. Tem de tomar as suas próprias decisões sobre o que fazer com eles.

2.1.11 Como lidar com requisitos contraditórios

Qualquer que seja a suite de sistema Debian que decida utilizar, pode ainda desejar correr versões de programas que não estão disponíveis nessa suite. Mesmo que encontre pacotes binários de tais programas noutras suites Debian ou noutros recursos não-Debian, os seus requisitos podem entrar em conflito com o seu sistema Debian atual.

Apesar de poder ajustar o sistema de gestão de pacotes com a técnica **apt-pinning**, etc., como descrito em Secção 2.7.7 para instalar esses pacotes binários dessincronizados, essas abordagens de ajuste têm apenas casos de utilização limitados uma vez que podem quebrar esses programas e o seu sistema.

Antes de instalar brutalmente tais pacotes fora de sincronia, você deve procurar todas as soluções técnicas alternativas mais seguras disponíveis que sejam compatíveis com o seu sistema Debian atual.

- Instale esses programas usando pacotes binários correspondentes em ”sandbox” (veja Secção 7.7).
  - Muitos programas GUI, como o LibreOffice e as aplicações GNOME, estão disponíveis como pacotes Flatpak, Snap, ou AppImage.
- Crie um ambiente chroot ou similar e execute esses programas nele (veja Secção 9.11).
  - Os comandos CLI podem ser executados facilmente sob o seu chroot compatível (ver Secção 9.11.4).
  - É possível experimentar facilmente vários ambientes de trabalho completos sem reiniciar (ver Secção 9.11.5).
- Construa você mesmo as versões desejadas dos pacotes binários que são compatíveis com o seu sistema Debian atual.
  - Trata-se de uma tarefa não trivial (ver Secção 2.7.13).

2.2 Operações básicas de gestão de pacotes

As operações de gestão de pacotes baseadas em repositório no sistema Debian podem ser executas por muitas ferramentas de gestão de pacotes baseadas no APT e disponíveis no sistema Debian. Aqui vamos explicar 3 ferramentas de gestão básica de pacotes: `apt`, `apt-get` / `apt-cache` e `aptitude`.

Para as operações de gestão de pacotes que envolvam a instalação ou atualização de meta-dados do pacote, necessita de ter privilégios de root.

2.2.1 `apt` vs. `apt-get` / `apt-cache` vs. `aptitude`

Apesar do `aptitude` ser uma ferramenta interactiva muito boa a qual o autor usa principalmente, deve ser advertido de alguns factos:

- O comando `aptitude` não é recomendado para a atualização de sistema de lançamento-a-lançamento do sistema Debian `stable` após um novo lançamento.
  - O uso de "`apt full-upgrade`" ou "`apt-get dist-upgrade`" é recomendado para isso. Veja Bug #411280.
- O comando `aptitude` por vezes sugere a remoção em massa de pacotes para a atualização do sistema no sistema Debian `testing` ou `unstable`.
  - Esta situação já assustou muitos administradores de sistemas. Não entre em pânico.
  - Isto parece ser causado principalmente pela torção de versões entre os pacotes que são dependências ou recomendações de um meta-pacote tal como o `gnome-core`.
  - Isto pode ser resolvido ao selecionar "Cancelar operações pendentes" no menu de comandos do `aptitude`, a terminar o `aptitude` e a usar "`apt full-upgrade`".

Os comandos `apt-get` e `apt-cache` são as ferramentas de gestão de pacotes baseadas no APT mais **básicas**.

- O `apt-get` e o `apt-cache` oferecem apenas a interface de linha de comandos.
- O `apt-get` é mais apropriado para uma **atualização maior ao sistema** entre lançamentos, etc.
- O `apt-get` oferece um resolvedor de dependências de pacotes **robusto**.
- `apt-get` é menos exigente em recursos de hardware. Consome menos memória e é mais rápido.
- O `apt-cache` oferece uma busca baseada em expressões regulares **standard** no nome do pacote e na descrição.
- O `apt-get` e o `apt-cache` podem gerir várias versões de pacotes a utilizar o `/etc/apt/preferences` mas é um pouco incómodo.

O comando `apt` é uma interface de linha de comandos de alto nível para gestão de pacotes. É basicamente um revestimento dos `apt-get`, `apt-cache` e comandos semelhantes, originalmente destinada a ser uma interface de utilizador final e ativa por predefinição algumas opções melhor apropriadas para utilização interativa.

- O `apt` disponibiliza uma barra de progresso amigável quando se instala pacotes a usar o `apt install`.
- O `apt` irá **remover** por predefinição os pacotes `.deb` em cache após instalação com sucesso dos pacotes descarregados.

Dica
É recomendado aos utilizadores usarem o novo comando apt(8) para uso **interativo** e usarem os comandos apt-get(8) e apt-cache(8) em script de shell.

O comando `aptitude` é a ferramenta de gestão de pacotes baseada no APT mais **versátil**.

- O `aptitude` oferece a interface de utilizador de texto interactiva de ecrã completo.
- O `aptitude` também oferece uma interface de utilizador de linha de comandos.
- O `aptitude` é mais apropriado para a **gestão de pacotes interactiva diária** como inspecionar os pacotes instalados e procurar pacotes disponíveis.
- O `aptitude` é mais exigente em recursos de hardware. Consome mais memória e é mais lento.
- O `aptitude` oferece um busca baseada em expressões regulares **avançada** em todos os meta-dados dos pacotes.
- O `aptitude` pode gerir várias versões de pacotes sem utilizar o `/etc/apt/preferences` e é bastante intuitivo.

2.2.2 Operações básicas de gestão de pacotes com a linha de comandos

Aqui estão algumas operações básicas de gestão de pacotes com a linha de comandos a usar apt(8), aptitude(8) e apt-get(8) /apt-cache(8).

`apt` / `apt-get` e `aptitude` podem ser misturados sem grandes problemas.

O "`aptitude why` *`expressão_regular`*" pode listar mais informação por "`aptitude -v why` *`expressão_regular`*". Informação semelhante pode ser obtida por `apt rdepends` *`pacote`* ou "`apt-cache rdepends` *`pacote`*".

Quando o comando `aptitude` é arrancado em modo de linha de comandos e enfrenta alguns problemas como conflitos de pacotes, pode mudar para modo interativo em ecrã total, ao pressionar a tecla "e", mais tarde na linha de comandos.

Nota
Apesar do comando `aptitude` vir com ricas funcionalidades como o resolvedor avançado de pacotes dele, esta complexidade já causou (ou pode ainda causar) algumas regressões como os Bug #411123, Bug #514930 e Bug #570377. Em caso de dúvidas, por favor utilize os comandos `apt`, `apt-get` e `apt-cache` em vez do comando `aptitude`.

Pode dar opções de comando logo após "`aptitude`".

Para mais veja aptitude(8) e o "Manual de utilizador do aptitude" em "`/usr/share/doc/aptitude/README`".

2.2.3 Uso interativo do aptitude

Para gestão de pacotes interativa, arranque o `aptitude` em modo interativo a partir da linha de comandos da consola conforme o seguinte:

```
$ sudo aptitude -u
Password:
```

Isto atualiza a cópia local da informação do arquivo e mostra a lista de pacotes em ecrã completo com menu. O aptitude coloca a configuração dele em "`~/.aptitude/config`".

Dica
Se desejar utilizar a configuração do root em vez da do utilizador, utilize "`sudo -H aptitude` …" em vez de "`sudo aptitude` …" na expressão acima.

Dica
O `aptitude` define automaticamente as **acções pendentes** como se fosse arrancado interativamente. Se não gostar disso, pode redefinir isto a partir do menu: "Acção" → "Cancelar acções pendentes".

---

sintaxe do `apt`	sintaxe do `aptitude`	sintaxe do `apt-get`/`apt-cache`	descrição
`apt update`	`aptitude update`	`apt-get update`	atualiza os meta-dados do arquivo de pacotes
`apt install foo`	`aptitude install foo`	`apt-get install foo`	instala a versão candidata do pacote "`foo`" com as suas dependências
`apt upgrade`	`aptitude safe-upgrade`	`apt-get upgrade`	instala as versões candidatas dos pacotes instalados sem remover quaisquer outros pacotes
`apt full-upgrade`	`aptitude full-upgrade`	`apt-get dist-upgrade`	instala as versões candidatas dos pacotes instalados a remover outros pacotes caso necessário
`apt remove foo`	`aptitude remove foo`	`apt-get remove foo`	remove o pacote "`foo`" a deixar os seus ficheiros de configuração
`apt autoremove`	N/D	`apt-get autoremove`	remove os pacotes auto-instalados que já não sejam necessários
`apt purge foo`	`aptitude purge foo`	`apt-get purge foo`	purga o pacote "`foo`" com os seus ficheiros de configuração
`apt clean`	`aptitude clean`	`apt-get clean`	limpa completamente o repositório local de ficheiros de pacotes obtidos
`apt autoclean`	`aptitude autoclean`	`apt-get autoclean`	limpa os pacotes desatualizados do repositório local dos ficheiros de pacotes recebidos
`apt show foo`	`aptitude show foo`	`apt-cache show foo`	mostra informação detalhada acerca do pacote "`foo`"
`apt search` *`regex`*	`aptitude search` *`regex`*	`apt-cache search` *`regex`*	procura pacotes que correspondem à *expressão-regular*
N/D	`aptitude why` *`regex`*	N/D	explica a razão porque o pacotes que correspondem à *expressão_regular* devem ser instalados
N/D	`aptitude why-not` *`regex`*	N/D	explica a razão porque o pacotes que correspondem à *expressão_regular* não podem ser instalados
`apt list --manual-installed`	`aptitude search '~i!~M'`	`apt-mark showmanual`	lista os pacotes instalados manualmente

Tabela 2.6: Operações básicas de gestão de pacotes com a linha de comandos a utilizar apt(8), aptitude(8) e apt-get(8) /apt-cache(8)

opção de comando	descrição
`-s`	simula o resultado do comando
`-d`	apenas descarrega e não instala/atualiza
`-D`	mostra breves explicações antes das instalações e remoções automáticas

Tabela 2.7: Opções de comando notáveis para o aptitude(8)

2.2.4 Teclas de atalho do aptitude

As combinações de teclas notáveis para explorar o estado dos pacotes e definir uma ”ação planeada” neles neste modo de ecrã total são as seguintes:

tecla	tecla de atalho
`F10` ou `Ctrl-t`	menu
`?`	mostra a **ajuda** para teclas (listagem mais completa)
`F10` → Ajuda → Manual do Utilizador	mostra o Manual do Utilizador
`u`	atualiza a informação de arquivo do pacote
`+`	marca o pacote para **atualização** ou **instalação**
`-`	marca o pacote para **remoção** (manter os ficheiros de configuração)
`_`	marca o pacote para **purgar** (remover ficheiros de configuração)
`=`	coloca o pacote em **retenção (hold)**
`U`	marca todos os pacotes com atualizações (funciona como **full-upgrade**)
`g`	começa a **descarregar** e a **instalar** os pacotes selecionados
`q`	sai do ecrã atual e guarda as alterações
`x`	sai do ecrã atual e descarta as alterações
`Enter`	ver informação acerca de um pacote
`C`	ver o relatório de alterações de um pacote
`l`	altera o limite dos pacotes mostrados
`/`	procura pela primeira correspondência
`\`	repetir a última pesquisa

Tabela 2.8: Lista de teclas de atalho do aptitude

A especificação de nome de ficheiro da linha de comandos ou do aviso de menu após pressionar ”`l`” e ”`//`” toma a expressão regular do aptitude conforme descrito em baixo. A expressão regular do aptitude pode corresponder explicitamente a um nome de pacote a utilizar uma cadeia começada por ”~n” e seguida do nome do pacote.

Dica
Necessita pressionar ”U”, no interface visual, para ter todos os pacotes instalados atualizados para a **versão candidata**. Caso contrário, apenas os pacotes selecionados e certos pacotes com dependências deles, versionadas, são atualizados à **versão candidata**.

2.2.5 Vistas de pacote no aptitude

No modo de ecrã completo interativo do aptitude(8), os pacotes na lista de pacotes são mostrados como no próximo exemplo.

```
idA   libsmbclient                              -2220kB 3.0.25a-1  3.0.25a-2
```

Aqui, esta linha significa desde a esquerda o seguinte:

- A flag ”estado atual” (a primeira letra)
- A flag ”acção planeada” (a segunda letra)
- A flag ”automático” (a terceira letra)
- O nome do Pacote
- A alteração na utilização do espaço do disco atribuída a ”acção planeada”

- A versão atual do pacote
- A versão candidata do pacote

Dica
A lista completa de flags é fornecida ao fundo do ecrã de **Ajuda** mostrada ao pressionar "?".

A **versão candidata** é escolhida de acordo com as preferências locais atuais (veja apt_preferences(5) e Secção 2.7.7).

Estão disponíveis vários tipos de vistas de pacotes sob o menu "`Vistas`".

vista	descrição da vista
`Package View`	veja Tabela 2.10 (predefinição)
`Audit Recommendations`	lista pacotes que são recomendados por alguns pacotes instalados mas ainda não estão instalados
`Flat Package List`	lista pacotes sem categorização (para utilizar com expressões regulares)
`Debtags Browser`	lista pacotes categorizados de acordo com as suas entradas debtags
`Source Package View`	lista de pacotes agrupados por pacotes fonte

Tabela 2.9: Lista de vistas para o aptitude

Nota
Por favor ajude-nos a melhorar a etiquetagem de pacotes com debtags!

A "`Vista de Pacotes` standard categoriza os pacotes de certo modo como o `dselect` com algumas funcionalidades extra.

categoria	descrição da vista
`Upgradable Packages`	lista pacotes organizados como `secção` → `área` → `pacote`
`New Packages`	,,
`Installed Packages`	,,
`Not Installed Packages`	,,
`Obsolete and Locally Created Packages`	,,
`Virtual Packages`	lista pacotes com a mesma função
`Tasks`	lista pacotes com diferentes funções geralmente necessárias para uma tarefa

Tabela 2.10: A categorização das vista de pacotes standard

Dica
A vista `Tarefas` pode ser usada para escolher pacotes para a sua tarefa.

2.2.6 Opções do método de pesquisa com o aptitude

O aptitude oferece várias opções para procurar pacotes a utilizar a fórmula de expressões regulares dele.

- Linha de comandos da shell:
  - "`aptitude search '`*`aptitude_regex`*`'`" para listar estado de instalação, nome do pacote e descrição curta dos pacotes correspondentes
  - "`aptitude show '`*`package_name`*`'`" para listar a descrição detalhada do pacote
- modo de ecrã total interativo:
  - "`l`" para limitar a vista de pacotes aos pacotes correspondentes
  - "`/`" para procurar um pacote correspondente
  - "`\`" para procurar um pacote correspondente a voltar para trás
  - "`n`" para procurar o próximo
  - "`N`" para procurar o próximo (a andar para trás)

Dica
A cadeia para *nome_de_pacote* é tratada como a correspondência exata da cadeia para o nome do pacote a menos que seja iniciada explicitamente com "~" para ser uma fórmula de expressão regular.

2.2.7 A fórmula regex do aptitude

A fórmula de expressão regular do aptitude é estendida tipo mutt **ERE** (veja Secção 1.6.2) e o significado das extensões de regras de correspondência especial específicas do `aptitude` são as seguintes:

- A parte da expressão regular é a mesma **ERE** que aquela utilizada nas típicas ferramentas de texto tipo-Unix que utilizam "^", "`.*`", "$" etc. como o egrep(1), awk(1) e perl(1).
- A dependência *type* é uma de (dependências, pré-dependências, recomendações, sugestões, conflitos, substituições, fornecimentos), que específica o inter-relacionamento do pacote.
- O *type* de dependência predefinida é "depends".

Dica
Quando *regex_pattern* for uma string nula, coloca "`~T`" imediatamente após o comando.

Aqui estão alguns atalhos.

- "`~P`*`term`*" == "`~Dprovides:`*`term`*"
- "`~C`*`term`*" == "`~Dconflicts:`*`term`*"
- "…`~W term`" == "`(…|term)`"

Os utilizadores familiarizados com o `mutt` aprendem rápido, pois o mutt foi a inspiração para a sintaxe de expressão. Veja "PROCURAR, LIMITAR E EXPRESSÕES" no "Manual do Utilizador" "`/usr/share/doc/aptitude/README`".

Nota
Com a versão `lenny` do aptitude(8), a nova sintaxe de **formato longo** como a "`?broken`" pode ser utilizada para correspondência de expressões regulares no lugar da equivalente antiga dele de **formato curto** "~b". Agora o caractere de espaço " " é considerado como um caractere terminante de expressão regular em adição ao caractere til "~". Veja o "Manual do Utilizador" para a nova sintaxe de **formato longo**.

---

descrição da regra de correspondência extensa	**fórmula da expressão regular**
corresponde com o nome do pacote	`~n`*`regex_name`*
corresponde com a descrição	`~d`*`regex_description`*
corresponde com nome da tarefa	`~t`*`regex_task`*
corresponde com debtag	`~G`*`regex_debtag`*
corresponde com o maintainer	`~m`*`regex_maintainer`*
corresponde com secção do pacote	`~s`*`regex_section`*
corresponde com versão do pacote	`~V`*`regex_version`*
corresponde com arquivo	`~A{trixie,forky,sid}`
corresponde com origem	`~O{debian,…}`
prioridade da correspondência	`~p{extra,important,optional,required,standard}`
corresponde com pacotes essenciais	`~E`
corresponde com pacotes virtuais	`~v`
corresponde com pacotes novos	`~N`
corresponde com acções pendentes	`~a{install,upgrade,downgrade,remove,purge,hold,keep}`
corresponde com os pacotes instalados	`~i`
corresponde com pacotes instalados com marca **A** (pacotes instalados automaticamente)	`~M`
corresponde com pacotes instalados sem a marca **A** (pacotes selecionados pelo administrador)	`~i!~M`
corresponde com pacotes instalados e com atualizações disponíveis	`~U`
corresponde com pacotes removidos mas não purgados	`~c`
corresponde com pacotes removidos, purgados ou que podem-ser-removidos	`~g`
corresponde com pacotes que declaram dependências quebradas	`~b`
corresponde com pacotes que declaram dependências quebradas de *type*	`~B`*`type`*
corresponde a pacotes *pattern* que declaram dependência de *type*	`~D[`*`type`*`:]`*`pattern`*
corresponde a pacotes *pattern* que declaram dependência quebrada de *type*	`~DB[`*`type`*`:]`*`pattern`*
corresponde a pacotes para os quais o pacote que corresponde a *pattern* declara o *type* de dependência	`~R[`*`type`*`:]`*`pattern`*
corresponde a pacotes para os quais o pacote que corresponde a *pattern* declara o *type* de dependência quebrada	`~RB[`*`type`*`:]`*`pattern`*
corresponde com pacotes com os quais alguns pacotes instalados dependem	`~R~i`
corresponde com pacotes com os quais nenhum outro pacote instalado depende	`!~R~i`
corresponde com pacotes com os quais alguns pacotes instalados dependem ou recomendam	`~R~i|~Rrecommends:~i`
corresponde o pacote *pattern* com a versão filtrada	`~S filter `*`pattern`*
corresponde com todos os pacotes (true)	`~T`
não corresponde com nenhum pacote (false)	`~F`

Tabela 2.11: Lista da fórmula regex do aptitude

2.2.8 Resolução de dependências do aptitude

A seleção de um pacote no `aptitude` não puxa apenas os pacotes definidos na lista de "`Dependências:`" dele, mas também os definidos na lista "`Recomendados:`" se o menu "`F10` → Opções → Preferências → Manuseamento de dependências" assim estiver definido. Estes pacotes auto-instalados são removidos automaticamente sob o `aptitude` se não forem mais necessários.

A flag que controla o comportamento "auto install" co comando `aptitude` também pode ser manipulada a usar o comando apt-mark(8) do pacote `apt`.

2.2.9 Relatórios (logs) de atividade de pacotes

Pode verificar o histórico de atividade de pacotes nos ficheiros log.

ficheiro	conteúdo
`/var/log/dpkg.log`	Log da atividade de nível do `dpkg` para as atividades de todos os pacotes
`/var/log/apt/term.log`	Log da atividade genérica do APT
`/var/log/aptitude`	Log da atividade de comandos do `aptitude`

Tabela 2.12: Os ficheiros log para atividades de pacotes

Na realidade, não é muito fácil obter rapidamente uma compreensão significativa a partir destes logs. Veja Secção 9.3.9 para um modo mais fácil.

2.3 Exemplos de operações do aptitude

Aqui estão alguns exemplos de operações do aptitude(8).

2.3.1 Procurar pacotes interessantes

Pode procurar os pacotes que satisfaçam as suas necessidades com o `aptitude` a partir da descrição do pacote ou a partir da lista "Tarefas".

2.3.2 Listagem de pacotes com correspondência por expressão regular nos nomes de pacotes

O seguinte comando lista pacotes com regex a condizer com nomes de pacotes.

```
$ aptitude search '~n(pam|nss).*ldap'
p libnss-ldap - NSS module for using LDAP as a naming service
p libpam-ldap - Pluggable Authentication Module allowing LDAP interfaces
```

Isto dá muito jeito para encontrar o nome exato de um pacote.

2.3.3 Explorar com a correspondência de expressão regular

a expressão regular "`~dipv6`" na vista "Nova Lista de Pacotes Simples" com o aviso "`l`", limita a vista aos pacotes com a descrição correspondente e permite-lhe explorar interativamente a informação deles.

2.3.4 Purgar pacotes removidos definitivamente

Pode purgar todos os restantes ficheiros de configuração dos pacotes removidos.

Verifique os resultados do seguinte comando.

```
# aptitude search '~c'
```

Se julgar que os pacotes listados podem ser purgados, execute o seguinte comando:

```
# aptitude purge '~c'
```

Pode fazer o mesmo no modo interativo para um controle mais preciso.

Fornece a expressão regular "`~c`" na vista "Nova Vista de Pacotes" com a prompt "`l`". Isto limita a vista de pacotes apenas aos pacotes correspondentes à expressão regular, isto é, "removidos mas não purgados". Todos estes pacotes correspondentes a expressões regulares podem ser mostrados ao pressionar "`[`" nos cabeçalhos de nível de topo.

Depois pressione "`_`" em cabeçalhos de nível de topo tal como "Pacotes Não Instalados". Apenas os pacotes correspondentes à expressão regular sob o cabeçalho são marcados para serem purgados com isto. Pode excluir alguns pacotes a serem purgados ao pressionar "`=`" interativamente para cada um deles.

Esta técnica é muito útil e funciona com muitas outras teclas de comando.

2.3.5 Acertar o estado auto/manual de instalação

Aqui está como acertar o estado auto/manual de instalação dos pacotes (após usar um instalador de pacotes sem ser o aptitude e etc.).

1. Arranque o `aptitude` em modo interativo como root.
2. Escreva "`u`", "`U`", "`f`" e "`g`" para atualizar a lista de pacotes e atualizar os pacotes.
3. Escreva "`l`" para inserir o limite de visualização de pacotes aos "`~i(~R~i|~Recomendados:~i)`" e escreva "`M`" sobre "`Pacotes Instalados`" como auto-instalado.
4. Escreva "`l`" para inserir o limite de visualização de pacotes como "`~required|~pimportant|~pstandard|~E`" e escreva "`m`" sobre "`Pacotes Instalados`" como instalados manualmente.
5. Escreva "`l`" para inserir o limite de visualização de pacotes como "`~i!~M`" e remover pacotes não utilizados ao escrever "`-`" sobre cada um deles após expô-los ao escrever "`[`" sobre "`Pacotes Instalados`".
6. Escreva "`l`" para inserir o limite de amostragem de pacotes como "`~i`" depois escreva "`m`" sobre "`Tasks`" para marcar esses pacotes como instalados manualmente.
7. Termina o `aptitude`.
8. Inicie "`apt-get -s autoremove|less`" como root para verificar os que não são usados.
9. Reinicie o `aptitude` em modo interativo e marque os pacotes necessários como "`m`".
10. Reinicie o "`apt-get -s autoremove|less`" como root para verificar que o REMOVED contém apenas os pacotes esperados.
11. Arranque "`apt-get autoremove|less`" como root para auto-remover os pacotes não usados.

A opção "`m`" sobre "`Tasks`" é uma opção para prevenir situações de remoção de pacotes em massa no futuro.

2.3.6 atualização total ao sistema

Nota
Quando mover para um novo lançamento etc, deverá considerar fazer uma instalação limpa do novo sistema mesmo a saber que Debian é atualizável como descrito em baixo. Isto dá-lhe a hipótese de remover os lixos coleccionados e expõe-lhe a melhor combinação do pacotes mais recentes. É claro que deverá fazer uma cópia de segurança do sistema para um lugar seguro (veja Secção 10.2) antes de fazer isto. Recomendo fazer uma configuração de duplo arranque a usar partições diferentes para ter a transição mais suave.

Pode efetuar a atualização de todo o sistema para uma versão mais recente alterando os conteúdos da **lista de fontes** apontando para uma nova versão e executando o comando "`apt update; apt dist-upgrade`".

Para atualizar de `stable` para `testing` ou `unstable` durante o ciclo de lançamento `trixie`-as-`stable`, você substitui "`trixie`" no exemplo **da lista de fontes** de Secção 2.1.5 por "`forky`" ou "`sid`".

Na realidade, pode enfrentar algumas complicações devido a problemas com a transição de alguns pacotes, na maioria devido a dependências desses pacotes. Quanto maior a diferença da atualização, maior a probabilidade de ter grandes problemas. Para a transição da `stable` antiga à nova `stable` após o lançamento dele, pode ler as novas Notas de Lançamento dele e seguir o procedimento exacto descrito lá para minimizar problemas.

Quando decidir mover de `stable` para `testing` antes do lançamento formal dele, não existem Notas de Lançamento para o ajudar. A diferença entre `stable` e `testing` pode ter crescido bastante após o lançamento `stable` anterior e complicar a situação da atualização.

Deve dar passos de precaução para a atualização total enquanto recolhe a informação mais recente da lista de mail e a usar senso comum.

1. Leia as "Notas de Lançamento" anteriores.
2. Faça cópia de segurança a todo o sistema (especialmente dados e informação de configuração).
3. Tenha um meio de arranque à mão para o caso do gestor de arranque ficar danificado.
4. Informe os utilizadores do sistema com bastante antecedência.
5. Grave a atividade de atualização com o script(1).
6. Para prevenir a remoção aplique "unmarkauto" aos pacotes necessários, p.e., "`aptitude unmarkauto vim`", .
7. Minimize a quantidade de pacotes instalados para reduzir a hipótese de conflitos de pacotes, p.e., remova os pacotes da tarefas de ambiente de trabalho.
8. Remova o ficheiro "`/etc/apt/preferences`" (desativa **apt-pinning**).
9. Tente a atualização em passos inteligentes: `oldstable` → `stable` → `testing` → `unstable`.
10. Atualize **a lista de fontes** para apontar apenas para o novo arquivo e execute "`aptitude update`".
11. Instale, opcionalmente, os novos **pacotes de base** primeiro, ex., "`aptitude install perl`".
12. Corra o comando "`apt-get -s dist-upgrade`" para avaliar o impacto.
13. Corra o comando "`apt-get dist-upgrade`" em último lugar.



Cuidado
Não é sensato saltar grandes lançamentos de Debian quando se atualiza entre lançamentos `stable`.



Cuidado
Nas ”Notas de Lançamento” anteriores, GCC, Linux Kernel, initrd-tools, Glibc, Perl, a cadeia de ferramentas do APT, etc. necessitaram de alguma atenção especial para a atualização geral do sistema.



Cuidado
As ”Notas de Lançamento” podem não cobrir todos os casos possíveis. Se você alterar configurações de baixo nível, a sua próxima atualização pode correr muito mal e falhar com ”... falha de segmentação após atualização ...”.

Para atualizações diárias em `unstable`, veja Secção 2.4.3.

2.4 Operações de gestão avançada de pacotes

2.4.1 Operações de gestão avançada de pacotes com linha de comandos

Aqui está uma lista de outras operações de gestão de pacotes para as quais o `aptitude` é de demasiado alto nível ou faltam-lhe funcionalidades necessárias.

Nota
Para um pacote com a funcionalidade multi-arch, pode precisar de especificar o nome da arquitectura para alguns comandos. Por exemplo, use ”`dpkg -L libglib2.0-0:amd64`” para listar o conteúdo do pacote `libglib2.0-0` para a arquitectura `amd64`.



Cuidado
As ferramentas de pacotes de nível mais baixo como ”`dpkg -i ...`” e ”`debi ...`” deverão ser utilizadas com cuidado pelo administrador do sistema. Não tomam conta automaticamente das dependências de pacotes necessárias. As opções de linha de comandos do dpkg ”`--force-all`” e semelhantes (veja dpkg(1)) destinam-se apenas a serem utilizadas por especialistas. Utiliza-las sem o conhecimento total dos seus efeitos pode danificar completamente o seu sistema.

Por favor note o seguinte:

- Toda a configuração do sistema e comandos de instalação necessitam ser executados pelo root.
- A contrário do `aptitude`, que utiliza regex (veja Secção 1.6.2), os outros comandos de gestão de pacotes utilizam padrões como a shell glob (veja Secção 1.5.6).
- O apt-file(1), é disponibilizado pelo pacote `apt-file`, tem de correr previamente ”`apt-file update`”.
- O configure-debian(8) disponibilizado pelo pacote `configure-debian` corre o dpkg-reconfigure(8) como seu backend.
- O dpkg-reconfigure(8) corre scripts de pacote a utilizar o debconf(1) como o backend dele.
- Os comandos ”`apt-get build-dep`”, ”`apt-get source`” e ”`apt-cache showsrc`” requerem a entrada ”`deb-src`” na **lista de fontes**.
- Os dget(1), debuild(1) e debi(1) necessitam do pacote `devscripts`.
- Veja o procedimento de (re)empacotamento a utilizar ”`apt-get source`” em Secção 2.7.13.
- O comando `make-kpkg` necessita do pacote `kernel-package` (veja Secção 9.10).
- Para empacotamento em geral veja Secção 12.9.

comando	acção
`COLUMNS=120 dpkg -l` *`package_name_pattern`*	lista o estado de um pacote instalado para o relatório de bug
`dpkg -L` *`package_name`*	lista o conteúdo de um pacote instalado
`dpkg -L` *`package_name`* `| egrep '/usr/share/man/man.*/.+'`	lista os manuais para um pacote instalado
`dpkg -S` *`file_name_pattern`*	lista os pacotes instalados que condizem com o nome de ficheiro
`apt-file search` *`file_name_pattern`*	lista pacotes no arquivo que condizem com o nome de ficheiro
`apt-file list` *`package_name_pattern`*	lista os conteúdos dos pacotes que correspondem no arquivo
`dpkg-reconfigure` *`package_name`*	reconfigura o pacote exacto
`dpkg-reconfigure -plow` *`package_name`*	reconfigura o pacote exacto com as questões mais detalhadas
`configure-debian`	reconfigura pacotes a partir do menu de ecrã completo
`dpkg --audit`	faz auditoria ao sistema por pacotes parcialmente instalados
`dpkg --configure -a`	configura todos os pacotes parcialmente instalados
`apt-cache policy` *`binary_package_name`*	mostra a versão disponível, a prioridade e informação de arquivo de um pacote binário
`apt-cache madison` *`package_name`*	mostra a versão disponível e informação de arquivo de um pacote
`apt-cache showsrc` *`binary_package_name`*	mostra informação do pacote de código-fonte de um pacote binário
`apt-get build-dep` *`package_name`*	instala os pacotes necessários para compilar pacote
`aptitude build-dep` *`package_name`*	instala os pacotes necessários para compilar pacote
`apt-get source` *`package_name`*	descarrega código-fonte (do arquivo standard)
`dget` *`URL for dsc file`*	descarrega um pacote de código-fonte (de outro arquivo)
`dpkg-source -x` *`package_name_version-debian.revision`*`.dsc`	constrói uma árvore de código-fonte a partir de um conjunto de pacotes de código-fonte (”`*.orig.tar.gz`” e ”`*.debian.tar.gz`”/”`*.diff.gz`”)
`debuild binary`	compila pacote(s) a partir de uma árvore fonte local
`make-kpkg kernel_image`	compila um pacote de kernel a partir de uma árvore fonte de kernel
`make-kpkg --initrd kernel_image`	compila um pacote de kernel a partir de uma árvore fonte de kernel com initramfs activa
`dpkg -i` *`package_name_version-debian.revision_arch`*`.deb`	instalar um pacote local no sistema
`apt install /path/to/`*`package_filename`*`.deb`	instala um pacote local no sistema, entretanto tenta resolver as dependências automaticamente
`debi` *`package_name_version-debian.revision_arch`*`.dsc`	instala pacote(s) locais no sistema
`dpkg --get-selections '*' >selection.txt`	guarda a informação de estado de seleção a nível de pacotes do `dpkg`
`dpkg --set-selections <selection.txt`	define a informação de estado de seleção a nível de pacotes do `dpkg`
`echo` *`package_name`* `hold | dpkg --set-selections`	define o estado de seleção de pacote ao nível do `dpkg` para **hold** (equivalente a ”`aptitude hold` *`nome_do_pacote`*”)

Tabela 2.13: Lista de operações de gestão avançada de pacotes

2.4.2 Verificação dos ficheiros pacotes instalados

A instalação de `debsums` permite a verificação dos ficheiros dos pacotes instalados contra valores MD5sum do ficheiro "`/var/lib/dpkg/info/*.md5sums`" com debsums(1). Para saber como o MD5sum funciona veja Secção 10.3.5 .

Nota
Como a base de dados MD5sum pode ser adulterada por um intruso, o debsums(1) é uma ferramenta de segurança de utilização limitada. É bom apenas para verificar modificações locais pelo administrador ou danos devido a erros de media.

2.4.3 Salvaguardar para problemas de pacotes

Muito utilizadores preferem seguir o lançamento **testing** (or **unstable**) do sistema Debian pelas suas novas funcionalidades e pacotes. Isto torna o sistema permeável a bugs críticos dos pacotes.

A instalação do pacote `apt-listbugs` salvaguarda o seu sistema contra bugs críticos ao verificar automaticamente o Debian BTS por bugs críticos quando fizer atualizações com o sistema APT.

A instalação do pacote `apt-listchanges` disponibiliza notícias importantes de "`NEWS.Debian`" ao atualizar com o sistema APT.

2.4.4 Procurar nos meta-dados do pacote

Embora hoje em dia visitar o site Debian https://packages.debian.org/ facilite a busca nos meta-dados do pacote, vamos ver modos mais tradicionais.

Os comandos grep-dctrl(1), grep-status(1) e grep-available(1) podem ser utilizados para procurar qualquer ficheiro que tenha o formato geral de um ficheiro de controle de pacote Debian.

"`dpkg -S` *`padrão_de_nome_de_ficheiro`*" pode ser utilizado para procurar nomes de pacotes instalados pelo `dpkg`que contenham ficheiros com o nome correspondente. Mas isto não vê os ficheiros criados pelo script do responsável do pacote.

Se necessitar de fazer uma busca mais elaborada nos meta-dados do dpkg, necessita executar o comando "`grep -e padrão_de_expressão_regular *`" no diretório "`/var/lib/dpkg/info/`". Isto fá-lo procurar as palavras mencionadas nos scripts dos pacotes e nos textos de questões de instalação.

Se desejar procurar, recursivamente, as dependências de pacotes, deverá utilizar o apt-rdepends(8).

2.5 Os interiores da gestão de pacotes Debian

Vamos aprender como o sistema de gestão de pacotes Debian funciona internamente. Isto deverá ajudá-lo a criar a sua própria solução para alguns problemas com pacotes.

2.5.1 Meta dados do arquivo

Os ficheiros de meta-dados para cada distribuição são armazenados sob "`dist/`*`nome-de_código`*" em cada site mirror Debian, p.e., "`http://deb.debian.org/debian/`". A estrutura de arquivo dele pode ser explorada com um navegador web. Existem 6 tipos de meta-dados chave.

No arquivo recente, estes meta-dados são armazenados como ficheiros diferenciais e comprimidos para reduzir o tráfego de rede.

ficheiro	localização	conteúdo
`Release`	topo da distribuição	descrição do arquivo e informação de integridade
`Release.gpg`	topo da distribuição	ficheiro de assinatura para o ficheiro "`Release`" assinado com a chave do arquivo
`Contents-`*`architecture`*	topo da distribuição	lista de todos os ficheiros para todos os pacotes no arquivo pertinente
`Release`	topo de cada combinação de distribuição/área/arquitectura	descrição do arquivo utilizada para a regra do apt_preferences(5)
`Packages`	topo de cada combinação de distribuição/área/arquitectura-binário	`debian/control` concatenado para pacotes binários
`Sources`	topo de cada combinação de distribuição/área/fonte	`debian/control` concatenado para pacotes fonte

Tabela 2.14: O conteúdo dos meta dados do arquivo Debian

2.5.2 Ficheiro "Release" de nível de topo e autenticidade:

Dica
O ficheiro "`Release`" no nível de topo é usado para assinar o arquivo sob o sistema **secure APT**.

Cada suite do arquivo Debian tem um ficheiro "`Release`" no nível de topo, p.e., "`http://deb.debian.org/debian/dis`" como o seguinte:

```
Origin: Debian
Label: Debian
Suite: unstable
Codename: sid
Date: Sat, 14 May 2011 08:20:50 UTC
Valid-Until: Sat, 21 May 2011 08:20:50 UTC
Architectures: alpha amd64 armel hppa hurd-i386 i386 ia64 kfreebsd-amd64 kfreebsd-i386 mips ↩
    mipsel powerpc s390 sparc
Components: main contrib non-free
Description: Debian x.y Unstable - Not Released
MD5Sum:
 bdc8fa4b3f5e4a715dd0d56d176fc789 18876880 Contents-alpha.gz
 9469a03c94b85e010d116aeeab9614c0 19441880 Contents-amd64.gz
 3d68e206d7faa3aded660dc0996054fe 19203165 Contents-armel.gz
...
```

Nota
Aqui, pode encontrar a minha lógica de utilizar "suite" e "nome de código" em Secção 2.1.5. A "distribuição" é usada quando se refere a ambos "suite" e "nome de código". Todos os nomes de "áreas" do arquivo oferecidos pelo arquivo são listados sob "Componentes".

A integridade do ficheiro de nível superior "`Release`" é verificada por uma infraestrutura criptográfica denominada apt seguro, tal como descrito em apt-secure(8).

- O ficheiro de assinatura criptográfica "`Release.gpg`" é criado a partir do ficheiro "`Release`" de nível de topo autenticado e da chave secreta do arquivo Debian.

- As chaves públicas do arquivo Debian são instaladas localmente pelo último pacote `debian-archive-keyring`.
- O sistema **APT seguro** verifica automaticamente a integridade do ficheiro de nível superior descarregado "`Release`" criptograficamente através deste ficheiro "`Release.gpg`" e das chaves públicas do arquivo Debian instaladas localmente.
- A integridade de todos os ficheiros "`Packages`" e "`Sources`" é verificada a utilizar valores MD5sum do ficheiro "`Release`" de nível de topo. A integridade de todos os ficheiros de pacotes é verificada a utilizar valores MD5sum nos ficheiros "`Packages`" e "`Sources`" Veja debsums(1) e Secção 2.4.2.
- Como a verificação de assinatura criptográfica é um processo muito mais intenso para a CPU do que o cálculo de valor MD5sum, a utilização de valores MD5sum para cada pacote enquanto se utiliza assinatura criptográfica para o ficheiro "`Release`" de nível de topo disponibiliza boa segurança com desempenho (veja Secção 10.3).

Se a entrada da **lista de fontes** especificar a opção "`signed-by`", a integridade do seu ficheiro de nível superior "`Release`" descarregado é verificada usando a chave pública especificada. Isto é útil quando **a lista de fontes** contém arquivos não-Debian.

Dica
A utilização do comando apt-key(8) para a gestão de chaves APT está obsoleta.

Além disso, pode verificar manualmente a integridade do ficheiro "`Release`" com o ficheiro "`Release.gpg`" e a chave pública do arquivo Debian colocada em ftp-master.debian.org utilizando o `gpg`.

2.5.3 Ficheiros "Release" do nível de arquivo

Dica
Os ficheiros "`Release`" do nível de arquivo são utilizados para a regra do apt_preferences(5).

Existem ficheiros "`Release`" de nível de arquivo para todas as localizações de arquivo especificadas pela **lista de fontes**, como "`http://deb.debian.org/debian/dists/unstable/main/binary-amd64/Release`" ou "`http://deb.debian.org/debian/dists/sid/main/binary-amd64/Release`", como se segue.

```
Archive: unstable
Origin: Debian
Label: Debian
Component: main
Architecture: amd64
```



Cuidado
Para a estrofe "`Archive:`" são utilizados os nomes de suite ("`stable`", "`testing`" e "`unstable`", ...) no arquivo Debian enquanto que os nomes de código ("`trusty`", "`xenial`", "`artful`", ...) são utilizados no arquivo Ubuntu.

Para alguns arquivos, tais como `experimental` e `trixie-backports`, que contêm pacotes que não devem ser instalados automaticamente, existe uma linha extra, p.e., "`http://deb.debian.org/debian/dists/experimental/`" como a seguir.

```
Archive: experimental
Origin: Debian
Label: Debian
NotAutomatic: yes
Component: main
Architecture: amd64
```

Por favor note que para arquivos normais sem "`NotAutomatic: yes`", o valor Pin-Priority predefinido é 500, enquanto que para arquivos especiais com "`NotAutomatic: yes`", o valor Pin-Priority predefinido é 1 (veja apt_preferences(5 e Secção 2.7.7).

2.5.4 Obter os meta dados do pacote

Quando são utilizadas ferramentas APT, tais como `aptitude`, `apt-get`, `synaptic`, `apt-file`, `auto-apt`, ..., nós precisamos de atualizar as cópias locais dos meta dados que contêm a informação do arquivo Debian. Estas cópias locais têm os seguintes nomes de ficheiros correspondentes à `distribution` especificada , `area`, e nomes de `architecture` na **lista de fontes** (veja Secção 2.1.5).

- "`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuição`*`_Release`"
- "`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuição`*`_Release.gpg`"
- "`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuição_área`*`_binário-`*`arquitectura`*`_Pa`
- "`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuição_área`*`_fonte_Sources`"
- "`/var/cache/apt/apt-file/deb.debian.org_debian_dists_`*`distribuição`*`_Contents-`*`arquitectura`*`.g` (para o `apt-file`)

Os primeiros 4 tipos de ficheiros são partilhados por todos os comandos APT pertinentes e atualizados a partir da linha de comandos por "`apt-get update`" ou "`aptitude update`". Os meta dados "`Packages`" são atualizados se o "`deb`" for especificado na **lista de fontes**. Os meta dados "`Sources`" são atualizados se o "`deb-src`" for especificado na **lista de fontes**.

Os meta-dados "`Packages`" e "`Sources`" contêm a estrofe "`Filename:`" que aponta à localização de ficheiro dos pacotes binários e de código-fonte. atualmente, estes pacotes estão localizados sob a árvore de diretórios "`pool/`" para a transição melhorada através dos lançamentos.

As cópias locais dos meta-dados "`Packages`" podem ser pesquisadas interativamente com a ajuda do `aptitude`. O comando de procura especializada grep-dctrl(1) pode pesquisar as cópias locais dos meta-dados "`Packages`" e "`Sources`".

A cópia local dos meta-dados "`Contents-`*`arquitectura`*" pode ser atualizada pelo "`apt-file update`" e a localização dele é diferente dos outros 4. Veja apt-file(1). (O `auto-apt` utiliza localização diferente para a cópia local de "`Contents-`*`arquitectura`*`.gz`" por predefinição.)

2.5.5 O estado dos pacote para o APT

Além dos meta-dados obtidos remotamente, a ferramenta APT após o `lenny` armazena a informação de estado de instalação dela gerada localmente em "`/var/lib/apt/extended_states`" que é utilizada por todas as ferramentas do APT para seguirem todos os pacotes auto-instalados.

2.5.6 O estado de pacotes para o aptitude

Além aos meta-dados obtidos remotamente, o `aptitude` armazena a sua informação de estado de instalação gerada localmente em "`/var/lib/aptitude/pkgstates`" que é usada apenas pelo próprio.

2.5.7 Copias locais dos pacotes obtidos

Todos os pacotes obtidos remotamente através do mecanismo APT são armazenados em "`/var/cache/apt/archives`" até que sejam limpos.

Esta politica de limpeza de ficheiros de cache para o `aptitude` pode ser definida em "`Opções`" → "`Preferências`" e pode ser forçada pelo seu menu "`Limpar cache de pacotes`" ou "`Limpar ficheiros obsoletos`" em "`Acções`".

2.5.8 Nomes de ficheiros de pacotes Debian

Ficheiros de pacotes Debian têm estruturas de nomes particulares.

tipo de pacote	estrutura de nomes
O pacote binário (a.k.a deb)	*package-name_upstream-version-debian.revision_architec*
O pacote binário para debian-installer (a.k.a udeb)	*package-name_upstream-version-debian.revision_architec*
O pacote fonte (código-fonte atual)	*package-name_upstream-version-debian.revision*.orig.tar
O pacote de código-fonte 1.0 (alterações do Debian)	*package-name_upstream-version-debian.revision*.diff.gz
O pacote de código-fonte 3.0 (quilt) (alterações do Debian)	*package-name_upstream-version-debian.revision*.debian.ta
O pacote de código-fonte (descrição)	*package-name_upstream-version-debian.revision*.dsc

Tabela 2.15: A estrutura de nomes dos pacotes Debian

Dica
Aqui apenas são descritos formatos de pacote fonte básicos. Veja mais em dpkg-source(1).

componente do nome	caracteres utilizáveis (ERE regex)	existência
package-name	[a-z0-9][-a-z0-9.+]+	necessário
epoch:	[0-9]+:	opcional
upstream-version	[-a-zA-Z0-9.+:]+	necessário
debian.revision	[a-zA-Z0-9.+~]+	opcional

Tabela 2.16: Os caracteres utilizáveis para cada componente nos nomes de pacotes Debian

Nota
Pode verificar a ordem da versão de pacotes com o dpkg(1), p.e., "dpkg --compare-versions 7.0 gt 7.~pre1 ; echo $?".

Nota
O debian-installer (d-i) utiliza udeb como a extensão de ficheiro para o pacote binário dele em vez do normal deb. Um pacote udeb é um pacote deb despido que remove alguns conteúdos não essenciais como a documentação para poupar espaço enquanto relaxa os requisitos de política do pacote. Ambos os pacotes deb e udeb partilham a mesma estrutura de pacote. O "u" significa micro.

2.5.9 O comando dpkg

dpkg(1) é a ferramenta de mais baixo nível para a gestão de pacotes Debian. É muito poderosa e tem que ser utilizada com cuidado.

Enquanto instala o pacote chamado "*package_name*", o dpkg processa-o na seguinte ordem.

1. Desempacota o ficheiro deb (equivalente a "ar -x")
2. Executa "*nome_de_pacote*.preinst" a utilizar o debconf(1)

3. Instala o conteúdo do pacote no sistema (equivalente a "`tar -x`")
4. Executa "*`nome_de_pacote`*`.postinst`" a utilizar o debconf(1)

O sistema `debconf` disponibiliza interacção standard com o utilizador com suporte de I18N e L10N (Capítulo 8).

ficheiro	**descrição dos conteúdos**
`/var/lib/dpkg/info/`*`package_name`*`.conffiles`	lista de ficheiros de configuração. (modificável pelo utilizador)
`/var/lib/dpkg/info/`*`package_name`*`.list`	lista de ficheiros e diretórios instalados pelo pacote
`/var/lib/dpkg/info/`*`package_name`*`.md5sums`	lista de valores de hash MD5 para os ficheiros instalados pelo pacote
`/var/lib/dpkg/info/`*`package_name`*`.preinst`	script de pacote para ser executado antes da instalação do pacote
`/var/lib/dpkg/info/`*`package_name`*`.postinst`	script de pacote para ser executado após a instalação do pacote
`/var/lib/dpkg/info/`*`package_name`*`.prerm`	script de pacote para ser executado antes da remoção do pacote
`/var/lib/dpkg/info/`*`package_name`*`.postrm`	script de pacote para ser executado após a remoção do pacote
`/var/lib/dpkg/info/`*`package_name`*`.config`	script de pacote para o sistema `debconf`
`/var/lib/dpkg/alternatives/`*`package_name`*	a informação alternativa usada pelo comando `update-alternatives`
`/var/lib/dpkg/available`	a informação de disponibilidade para todos os pacotes
`/var/lib/dpkg/diversions`	a informação de diversões usadas pelo dpkg(1) e definidas por dpkg-divert(8)
`/var/lib/dpkg/statoverride`	a informação de sobreposição de estado usada pelo dpkg(1) e definida por dpkg-statoverride(8)
`/var/lib/dpkg/status`	a informação de estado para todos os pacotes
`/var/lib/dpkg/status-old`	o backup de primeira geração do ficheiro "`var/lib/dpkg/status`"
`/var/backups/dpkg.status*`	o backup de segunda geração e os mais antigos do ficheiro "`var/lib/dpkg/status`"

Tabela 2.17: Ficheiros notáveis criados pelo `dpkg`

O ficheiro "`status`" também é utilizado por ferramentas como o dpkg(1), o "`dselect update`" e o "`apt-get -u dselect-upgrade`".

O comando especializado de busca grep-dctrl(1) pode procurar as cópias locais dos meta dados "`status`" e "`available`".

Dica
No ambiente do debian-installer, o comando `udpkg` é usado para abrir pacotes `udeb`. O comando `udpkg` é uma versão reduzida do comando `dpkg`.

2.5.10 O comando update-alternatives

O sistema Debian tem um mecanismo para instalar programas de certa maneira sobrepostos de um modo pacífico a usar update-alternatives(1). Por exemplo, pode fazer o comando `vi` selecionar o `vim` para executar enquanto instala ambos os pacotes `vim` e `nvi`.

```
$ ls -l $(type -p vi)
lrwxrwxrwx 1 root root 20 2007-03-24 19:05 /usr/bin/vi -> /etc/alternatives/vi
$ sudo update-alternatives --display vi
...
$ sudo update-alternatives --config vi
  Selection    Command
 -----------------------------------------------
      1        /usr/bin/vim
```

```
*+    2        /usr/bin/nvi

Enter to keep the default[*], or type selection number: 1
```

O sistema de alternativas do Debian mantêm a sua seleção como ligação simbólica em "`/etc/alternatives/`". O processo de seleção utiliza um ficheiro correspondente em "`/var/lib/dpkg/alternatives/`".

2.5.11 O comando dpkg-statoverride

Stat overrides disponibilizados pelo comando dpkg-statoverride(8) são um modo de dizer ao dpkg(1) para usar um dono ou modo diferente para um **ficheiro** quando um pacote for instalado. Se for especificado "`--update`" e o ficheiro existir é imediatamente definido para o novo dono e modo.



Cuidado
A alteração directa do dono ou modo para um **ficheiro** cujo dono é o pacote a usar os comandos `chmod` ou `chown` pelo administrador do sistema é reiniciada pela próxima atualização do pacote.

Nota
Uso a palavra **ficheiro** aqui, mas na verdade pode ser qualquer objecto de sistema de ficheiros com que o `dpkg` lide, incluindo diretórios, dispositivos, etc.

2.5.12 O comando dpkg-divert

As **diversões** de ficheiros disponibilizadas pelo comando dpkg-divert(8) são um modo de forçar o dpkg(1) a não instalar um ficheiro na localização predefinida dele, mas para uma localização **divergida**. Os uso do `dpkg-divert` destina-se aos scripts do responsável do pacote. A utilização casual dele pelo administrador do sistema está descontinuada.

2.6 Recuperação de um sistema danificado

Quando corre o sistema `testing` ou `unstable`, espera-se que o administrador saiba recuperar o sistema de situações de gestão de pacotes com conflitos.



Cuidado
Alguns métodos descritos aqui são acções de alto risco. Foi avisado!

2.6.1 Incompatibilidade com configurações antigas de utilizador

Se um programa GUI de ambiente de trabalho ficou instável após uma atualização significante da versão original, deve suspeitar de interferências com os ficheiros locais de configuração antigos criados por ele. Se estiver estável sob uma nova conta de utilizador criada, esta hipótese está confirmada. (Isto é um bug de empacotamento e geralmente evitado pelo empacotador.)

Para recuperar a estabilidade, deve mover os ficheiros de configuração locais correspondentes e reiniciar o programa GUI. Poderá ter que ler o conteúdo dos ficheiros de configuração antigos para mais tarde recuperar informação de configuração. (Não os apague muito depressa.)

2.6.2 Erros de armazenamento em cache dos dados do pacote

Erros de cache dos dados do pacote causam erros intrigantes, tais como "Erro GPG: ... invalid: BADSIG ..." com APT.

Deve remover todos os dados guardados em cache com "`sudo rm -rf /var/lib/apt/*` " e tentar novamente. (Se for utilizado `apt-cacher-ng`, deve também correr "`sudo rm -rf /var/cache/apt-cacher-ng/*` ".)

2.6.3 Recuperação com o comando dpkg

Como o `dpkg` é uma ferramenta de pacotes de muito baixo nível, consegue funcionar sem uma ligação à rede.

Vamos assumir que o pacote `foo` está quebrado e precisa ser corrigido.

Pode ainda encontrar cópias em cache de uma versão antiga livre de bugs do pacote `foo` no diretório de cache de pacotes: "`/var/cache/apt/archives/`". (se não, pode descarregá-lo do arquivo https://snapshot.debian.org/ ou copiá-lo da cache de pacotes de uma máquina funcional.)

Se puder arrancar o sistema, pode instalá-lo com o seguinte comando.

```
# dpkg -i /path/to/foo_old_version_arch.deb
```

Se a tentativa de instalar um pacote deste modo falha devido a algumas violações de dependências e necessitar realmente de fazer isto como último recurso, pode sobrepor a dependência a utilizar "`--ignore-depends`", "`--force-depends`" e outras opções do `dpkg`. Se o fizer, precisa de fazer um sério esforço para restaurar as dependências apropriadas mais tarde. Veja dpkg(8) para mais detalhes.

Nota
Se o seu sistema estiver seriamente danificado, deve fazer uma salvaguarda completa para um lugar seguro (veja Secção 10.2) e deve fazer uma instalação limpa. Isto consome menos tempo e produz melhores resultados no fim.

Dica
Se os danos no sistema forem menores, pode em alternativa fazer um downgrade (regredir a versão) ao sistema completo como em Secção 2.7.11 a usar o nível mais alto do sistema APT.

2.6.4 Falha na instalação devido a dependências em falta

Se forçar a instalação de um pacote através de "`sudo dpkg -i ...`" num sistema sem todos os pacotes de dependência instalados, a instalação do pacote irá falhar como parcialmente instalado.

Deve instalar todos os pacotes de dependência usando repetidamente "`sudo dpkg -i ...`" ou usando:

```
# apt --fix-broken install
```

Em seguida, configure todos os pacotes parcialmente instalados com o seguinte comando.

```
# dpkg --configure -a
```

2.6.5 Pacotes diferentes com ficheiros sobrepostos

Os sistemas de gestão de pacotes a nível de arquivo, como o aptitude(8) ou o apt-get(1), nem tentam instalar pacotes com ficheiros sobrepostos a utilizar as dependências do pacote. (veja Secção 2.1.7).

Erros do responsável do pacote ou de implantação inconsistente de mistura de fontes de arquivos (veja Secção 2.7.6) pelo administrador do sistema podem criar situações com dependências de pacotes definidas incorrectamente. Quando instala um pacote com ficheiros sobrepostos a usar o aptitude(8) ou o apt-get(1) sob tal situação, o dpkg(1) que desempacota o pacote certifica-se de retornar um erro ao programa que o chama sem sobrescrever os ficheiros existentes.



Cuidado
A utilização de pacotes de terceiros introduz riscos significantes ao sistema através dos scripts do programador do pacote que são executados com privilégios de root e podem fazer o que quiserem ao seu sistema. O comando dpkg(1) apenas protege contra a sobreposição de ficheiros ao desempacotar.

Pode contornar tal problema de instalação ao remover primeiro o pacote ofensivo antigo, *pacote_antigo*.

```
$ sudo dpkg -P old-package
```

2.6.6 Corrigir script problemático de pacote

Quando um comando no script do pacote retorna erro por alguma razão e o script termina com erro, o sistema de gestão de pacotes aborta a acção dele e termina com pacotes parcialmente instalados. Quando um pacote contém bugs nos seus scripts de remoção, o pacote pode tornar-se impossível de remover e isso é bastante desagradável.

Para o problema script de pacote "*package_name*", deve observar os seguintes scripts do pacote.

- "`/var/lib/dpkg/info/`*nome_do_pacote*`.preinst`"
- "`/var/lib/dpkg/info/`*nome_do_pacote*`.postinst`"
- "`/var/lib/dpkg/info/`*nome_do_pacote*`.prerm`"
- "`/var/lib/dpkg/info/`*nome_do_pacote*`.postrm`"

Editar o script do pacote ofensivo a partir de root a usar as seguintes técnicas:

- desativar a linha ofensiva ao preceder um "#"
- forçar um retorno com sucesso ao acrescentar a linha ofensiva com "`|| true`"

Em seguida, configure todos os pacotes parcialmente instalados com o seguinte comando.

```
# dpkg --configure -a
```

2.6.7 Recuperar dados de seleção de pacotes

Se por qualquer razão o "`/var/lib/dpkg/status`" ficar corrompido o sistema Debian perde os dados de seleção de pacotes e sofre severamente. Procure o ficheiro antigo "`/var/lib/dpkg/status`" em "`/var/lib/dpkg/status-old`" ou "`/var/backups/dpkg.status.*`".

Manter "`/var/backups/`" numa partição separada pode ser uma boa ideia porque este diretório contém muitos dados importantes do sistema .

Em caso de sérios danos recomendo fazer uma instalação limpa após fazer a salvaguarda do sistema. Mesmo que tudo em "`/var/`" esteja perdido, ainda pode recuperar alguma informação dos diretórios em "`/usr/share/doc/`" para guiar a sua nova instalação.

Reinstalar o sistema mínimo (ambiente de trabalho).

```
# mkdir -p /path/to/old/system
```

Monte o sistema antigo em "`/caminho/para/sistema/antigo/`".

```
# cd /path/to/old/system/usr/share/doc
# ls -1 >~/ls1.txt
# cd /usr/share/doc
# ls -1 >>~/ls1.txt
# cd
# sort ls1.txt | uniq | less
```

Então ser-lhe-ão apresentados nomes de pacotes para instalar. (Podem existir alguns nomes que não de pacotes como "`texmf`".)

2.7 Dicas para a gestão de pacotes

Para simplificar, os exemplos de **listas de fontes** nesta secção são apresentados como "`/etc/apt/sources.list`" em estilo de uma linha após o lançamento do `trixie`.

2.7.1 Quem fez o upload do pacote?

Apesar do nome do responsável listado em "`/var/lib/dpkg/available`" e "`/usr/share/doc/package_name/chang`" fornecer alguma informação sobre "quem está por detrás a atividade de empacotamento", quem faz o upload real do pacote é um tanto obscuro. O who-uploads(1) no pacote `devscripts` identifica quem foi o uploader real dos pacotes fonte Debian.

2.7.2 Limitar a largura de banda de descarga para o APT

Se desejar limitar a largura de banda para o APT a, por exemplo, 800Kib/sec (=100kiB/sec), deve configurar o APT e o parâmetro de configuração dele conforme o seguinte.

```
APT::Acquire::http::Dl-Limit "800";
```

2.7.3 Descarga e atualização automática de pacotes

O pacote `apt` vem com um script de cron próprio "`/etc/cron.daily/apt`" para suportar a descarga automática de pacotes. Este script pode ser melhorado para executar a atualização automática de pacotes ao instalar o pacote `unattended-upgrades`. Esta pode ser personalizada por parâmetros em "`/etc/apt/apt.conf.d/02backup`" e "`/etc/apt/apt.conf.d/50unattended-upgrades`" conforme descrito em "`/usr/share/doc/unattended-upgr`

O pacote `unattended-upgrades` destina-se principalmente às atualizações de segurança do sistema `stable`. Se o risco de danificar um sistema `stable` existente pelas atualizações automáticas for menor que ser danificado por um intruso que usa buracos de segurança que foram fechados por atualizações de segurança, deve considerar usar estas atualizações automáticas com parâmetros de configuração como os a seguir.

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "1";
APT::Periodic::Unattended-Upgrade "1";
```

Se está a correr um sistema `testing` ou `unstable`, não quer utilizar a atualização automática pois certamente irá quebrar o sistema um dia. Mesmo para esse caso `testing` ou `unstable`, pode querer descarregar pacotes antecipadamente para poupar tempo para a atualização interativa com parâmetros de configuração como os seguintes.

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "1";
APT::Periodic::Unattended-Upgrade "0";
```

2.7.4 Atualizações e Backports

Existem stable-updates ("`trixie`-updates" durante o ciclo de lançamento `trixie`-como-`stable`) e arquivos backports.debian.org que disponibilizam pacotes de atualização para `stable`.

De modo a utilizar estes arquivos, liste todos os arquivos necessários no ficheiro "`/etc/apt/sources.list`" como a seguir:

```
deb http://deb.debian.org/debian/ trixie main non-free-firmware contrib non-free
deb http://security.debian.org/debian-security trixie-security main non-free-firmware ↩
    contrib non-free
deb http://deb.debian.org/debian/ trixie-updates main non-free-firmware contrib non-free
deb http://deb.debian.org/debian/ trixie-backports main non-free-firmware contrib non-free
```

Não há necessidade de definir valores específicos de Pin-Priority no ficheiro "`/etc/apt/preferences`". Quando os novos pacotes ficam disponíveis, a configuração predefinida disponibiliza as atualizações mais razoáveis (veja Secção 2.5.3).

- Todos os pacotes antigos instalados são atualizados para mais recentes a partir de `trixie-updates`.
- Apenas os pacotes antigos instalados manualmente a partir de `trixie-backports` são atualizados para mais recentes a partir de `trixie-backports`.

Sempre que desejar instalar um pacote chamado "*`nome-do-pacote`*" com as suas dependências a partir do arquivo `trixie-backports` manualmente, utilize o seguinte comando enquanto muda o lançamento alvo com a opção "`-t`".

```
$ sudo apt-get install -t trixie-backports package-name
```



Atenção
Não instale demasiados pacotes dos arquivos backports.debian.org. Isso pode causar complicações na dependência de pacotes. Veja Secção 2.1.11 as soluções alternativas.

2.7.5 Arquivos de pacotes externos



Atenção
Deve ter em atenção que o pacote externo ganha o privilégio de root no seu sistema. Só deve utilizar o arquivo de pacotes externos de confiança. Ver Secção 2.1.11 para soluções alternativas.

Pode utilizar o APT seguro com um arquivo de pacotes externo compatível com Debian adicionando-o à **lista de fontes** e ao seu ficheiro de chave de arquivo no diretório "`/etc/apt/trusted.gpg.d/`". Veja sources.list(5), apt-secure(8) e apt-key(8).

2.7.6 Pacotes de fontes mistas de arquivos sem apt-pinning



Cuidado
Instalar pacotes de fontes misturadas de arquivos não é suportado pela distribuição oficial Debian excepto para combinações de arquivos oficialmente suportadas tais como a `stable` com security updates e stable-updates.

Aqui está um exemplo de operações para incluir, uma vez, pacotes específicos com novas versões da origem encontrados em `unstable` enquanto se acompanha a `testing`.

1. Altere o ficheiro "`/etc/apt/sources.list`" temporariamente para entrada única "`unstable`".
2. Corra "`aptitude update`".
3. Corra "`aptitude install` *`nome-do-pacote`*".
4. Recupere o ficheiro "`/etc/apt/sources.list`" original para `testing`.
5. Corra "`aptitude update`".

Não crie o ficheiro "`/etc/apt/preferences`" nem precisa de se preocupar com o **apt-pinning** com esta abordagem manual. Mas isto é muito incómodo.



Cuidado
Quando utiliza fontes misturadas de arquivos, tem que assegurar por si próprio a compatibilidade dos pacotes pois Debian não o garante. Se existir incompatibilidade de pacotes, pode danificar o seu sistema. Tem que ser capaz de julgar estes requisitos técnicos. A utilização de fontes misturadas de arquivos aleatórios é uma operação completamente opcional e a utilização deles não é algo que o encoraje a utilizar.

As regras gerais para instalar pacotes de arquivos diferentes são as seguintes.

- Pacotes não-binários de ("`Architecture: all`") são **mais seguros** para instalar.
  - pacotes de documentação: sem requisitos especiais
  - pacotes de programa interpretador: tem de estar disponível interpretador compatível
- Pacotes binários (não "`Architecture: all`") geralmente enfrentam muitos obstáculos e são **inseguros** para instalar.
  - compatibilidade de versão de biblioteca (incluindo a "`libc`")
  - compatibilidade de versão de programa utilitário relacionada
  - compatibilidade da ABI do Kernel
  - Compatibilidade ABI C++
  - …

Nota
De modo a tornar um pacote **seguro** para instalar, alguns pacotes de programas binários comerciais não-livres podem vir fornecidos com bibliotecas completamente ligadas estaticamente. Mesmo assim deve verificar problemas de compatibilidade da ABI e etc. com eles.

Nota
Exceto para evitar pacotes partidos por um curto período de tempo, instalar pacotes binários de repositórios não-Debian é geralmente má ideia. Você deve procurar todas as soluções técnicas alternativas mais seguras disponíveis que sejam compatíveis com o seu sistema Debian atual (veja Secção 2.1.11).

2.7.7 Ajustar a versão candidata com o apt-pinning



Atenção
A utilização da técnica **apt-pinning** por um utilizador novato irá certamente causar grandes problemas. Deve evitar utilizar esta técnica exceto quando for absolutamente necessário. Veja Secção 2.1.11 para soluções alternativas.

Sem o ficheiro "`/etc/apt/preferences`", o sistema APT escolhe a versão disponível mais recente com a **versão candidata** a utilizar a cadeia de versão. Este é o estado normal e a utilização recomendada do sistema APT. Todas as combinações de arquivos oficialmente suportadas não necessitam do ficheiro "`/etc/apt/preferences`" porque alguns arquivos que não devem ser utilizados como fonte automática de atualizações são marcados como **NotAutomatic** e são tratados de modo apropriado.

Dica
A regra de comparação da string de versão pode ser verificada com, p.e., "`dpkg --compare-versions ver1.1 gt ver1.1~1; echo $?`" (veja dpkg(1)).

Quando instala regularmente pacotes de uma mistura de fontes de arquivos (veja Secção 2.7.6), pode automatizar estas operações complicadas ao criar o ficheiro "`/etc/apt/preferences`" com entradas apropriadas e a moldar a regra de seleção de pacotes para a **versão candidata** como descrito em apt_preferences(5). Isto chama-se **apt-pinning**.

Quando utilizar **apt-pinning**, você próprio tem que assegurar a compatibilidade dos pacotes pois Debian não o garante. O **apt-pinning** é uma operação completamente opcional e a sua utilização não é algo que se encoraje.

Os níveis de arquivo dos ficheiros de lançamento (veja Secção 2.5.3) são utilizados para a regra do apt_preferences(5). Assim o **apt-pinning** funciona apenas com nome de "suite" para arquivos Debian normais e arquivos Debian de segurança. (Isto é diferente dos arquivos do Ubuntu.) Por exemplo, pode fazer "`Pin: release a=unstable`" mas não pode fazer "`Pin: release a=sid`" no ficheiro "`/etc/apt/preferences`".

Quando utilizar um arquivo não-Debian como parte de **apt-pinning**, deve verificar ao que ele se destina e também verificar a credibilidade dele. Por exemplo, Ubuntu e Debian não se destinam a ser misturados.

Nota
Mesmo que não crie o ficheiro "`/etc/apt/preferences`", pode fazer operações no sistema bastante complexas (veja Secção 2.6.3 e Secção 2.7.6) sem o **apt-pinning**.

Aqui está uma explicação simplificada da técnica de **apt-pinning**.

O sistema APT escolhe o pacote de **atualização** com o Pin-Priority maior das fontes de pacotes disponíveis definidas no ficheiro "`/etc/apt/sources.list`" como o pacote de **versão candidata**. Se o Pin-Priority do pacote for maior que 1000, esta restrição de versão para **atualização** é abandonada para permitir a regressão (veja Secção 2.7.11).

O valor Pin-Priority de cada pacote é definido por entradas "Pin-Priority" no ficheiro "`/etc/apt/preferences`" ou utiliza o valor predefinido dele.

O arquivo da **suite alvo** pode ser definido pela linha de comando, por exemplo, "`apt-get install -t testing some-package`"

Os arquivos **NotAutomatic** e **ButAutomaticUpgrades** são definidos pelo servidor de arquivo que contêm no ficheiro Release dele do nível de arquivo (veja Secção 2.5.3) ambos "`NotAutomatic: yes`" e "`ButAutomaticUpgrades: yes`". O arquivo **NotAutomatic** é definido pelo servidor de arquivo que contém no ficheiro Release de nível de arquivo dele apenas "`NotAutomatic: yes`".

A **situação de apt-pinning** do *pacote* de várias fontes de arquivos é mostrada por "`apt-cache policy pacote`".

Prioridade-pin	efeitos do apt-pinning no pacote
1001	instala o pacote mesmo que isto constitua uma regressão na versão (downgrade) do pacote
990	utilizado como predefinição para o arquivo de **lançamento de destino**
500	utilizado por predefinição para o arquivo **normal**
100	utilizado como predefinição para os arquivos **NotAutomatic** e **ButAutomaticUpgrades**
100	utilizado para o pacote **instalado**
1	utilizado como predefinição para o arquivo **NotAutomatic**
-1	**nunca instala** o pacote mesmo que este seja recomendado

Tabela 2.18: Lista de valores notáveis de Pin-Priority para a técnica de **apt-pinning**.

- Uma linha começada com "`Package pin:`" lista a versão do pacote de **pin** se estiver definida a associação apenas com o *pacote* p.e., "`Package pin: 0.190`".
- Não existe nenhuma linha com "`Package pin:`" se não estiver definida nenhuma associação apenas com *pacote*.
- O valor Pin-Priority a associar ao *pacote* é listado no lado direito de todas as strings de versão, p.e., "`0.181 700`".
- É listado "`0`" à direita de todas as strings de versão se nenhuma associação apenas com *pacote* for definida, ex., "`0.181 0`".
- Os valores Pin-Priority dos arquivos (definidos como "`Package: *`" no ficheiro "`/etc/apt/preferences`") são listados à esquerda dos caminhos dos arquivos, ex., "`100 http://deb.debian.org/debian/ trixie-backports Packages`".

2.7.8 Bloquear pacotes instalados por "Recomendados"



Atenção
A utilização da técnica **apt-pinning** por um utilizador novato irá certamente causar grandes problemas. Deve evitar utilizar esta técnica exceto quando for absolutamente necessário. Veja Secção 2.1.11 para soluções alternativas.

Se desejar não puxar determinados pacotes automaticamente através de "Recommends", tem de criar o ficheiro "`/etc/apt/preferences`" e listar explicitamente esses pacotes no topo conforme a seguir:

```
Package: package-1
Pin: version *
Pin-Priority: -1

Package: package-2
Pin: version *
Pin-Priority: -1
```

2.7.9 Acompanhar `testing` com alguns pacotes de `unstable`



Atenção
A utilização da técnica **apt-pinning** por um utilizador novato irá certamente causar grandes problemas. Deve evitar utilizar esta técnica exceto quando for absolutamente necessário. Veja Secção 2.1.11 para soluções alternativas.

Aqui está um exemplo de técnica de **apt-pinning** para incluir pacotes específicos de versão original mais recente encontrados em `unstable` e atualizados regularmente enquanto segue a suite `testing`. Liste todos os arquivos necessários no ficheiro "`/etc/apt/sources.list`" conforme a seguir:

```
deb http://deb.debian.org/debian/ testing main contrib non-free
deb http://deb.debian.org/debian/ unstable main contrib non-free
deb http://security.debian.org/debian-security testing-security main contrib
```

Configure o ficheiro "`/etc/apt/preferences`" como o seguinte:

```
Package: *
Pin: release a=unstable
Pin-Priority: 100
```

Quando desejar instalar um pacote chamado "*`nome_do_pacote`*" com as suas dependências a partir do arquivo `unstable` sob esta configuração, invoque o seguinte comando que muda o lançamento alvo com a opção "`-t`" (o Pin-Priority de `unstable` torna-se 990).

```
$ sudo apt-get install -t unstable package-name
```

Com esta configuração, a execução usual de "`apt-get upgrade`" e "`apt-get dist-upgrade`" (ou "`aptitude safe-upgrade`" e "`aptitude full-upgrade`") atualiza os pacotes que foram instalados a partir do repositório `testing` ao usar a suite `testing` atual e os pacotes que foram instalados a partir do repositório `unstable` a usar a suite `unstable` atual.



Cuidado
Tenha cuidado para não remover a entrada "`testing`" do ficheiro "`/etc/apt/sources.list`". Sem a entrada "`testing`" lá, o sistema APT atualiza os pacotes do novo arquivo `unstable`.

Dica
Geralmente edito o ficheiro "`/etc/apt/sources.list`" para comentar a entrada do arquivo "`unstable`" logo após a operação acima. Isto evita a lentidão do processo de atualização ao ter demasiadas entradas no ficheiro "`/etc/apt/sources.list`" embora isto impeça a atualização dos pacotes que foram instalados a partir do arquivo `unstable` a utilizar o arquivo `unstable` atual.

Dica
Se for utilizado "`Pin-Priority: 1`" em vez de "`Pin-Priority: 100`" no ficheiro "`/etc/apt/preferences`", os pacotes já instalados que têm o valor Pin-Priority de 100 não são atualizados pelo arquivo `unstable` mesmo se a entrada "`testing`" no ficheiro "`/etc/apt/sources.list`" seja removida.

Se desejar acompanhar pacotes particulares em `unstable` automaticamente sem uma instalação inicial "`-t unstable`", tem de criar o ficheiro "`/etc/apt/preferences`" e listar explicitamente todos esses pacotes no topo conforme a seguir:

```
Package: package-1
Pin: release a=unstable
Pin-Priority: 700

Package: package-2
Pin: release a=unstable
Pin-Priority: 700
```

Estes definem o valor Pin-Priority para cada pacote específico. Por exemplo, de modo a acompanhar a versão `unstable` mais recente deste "Debian Reference" em Português, deve ter as seguintes entradas no ficheiro "`/etc/apt/pr`

```
Package: debian-reference-en
Pin: release a=unstable
Pin-Priority: 700

Package: debian-reference-common
Pin: release a=unstable
Pin-Priority: 700
```

Dica
Esta técnica de **apt-pinning** é válida mesmo se estiver a seguir o arquivo `stable`. Pela minha experiência e até agora, os pacotes de documentação sempre foram seguros de instalar a partir do arquivo `unstable`.

2.7.10 Acompanhar `unstable` com alguns pacotes de `experimental`



Atenção
A utilização da técnica **apt-pinning** por um utilizador novato irá certamente causar grandes problemas. Deve evitar utilizar esta técnica exceto quando for absolutamente necessário. Veja Secção 2.1.11 para soluções alternativas.

Aqui está outro exemplo de técnica de **apt-pinning** para incluir pacotes de versão de origem mais recentes encontrados em `experimental` enquanto segue `unstable`. Lista todos os arquivos necessários no ficheiro "`/etc/apt/sources.list`" conforme o seguinte:

```
deb http://deb.debian.org/debian/ unstable main contrib non-free
deb http://deb.debian.org/debian/ experimental main contrib non-free
deb http://security.debian.org/ testing-security main contrib
```

O valor Pin-Priority predefinido para o arquivo `experimental` é sempre 1 (<<100) porque é um arquivo **NotAutomatic** (veja Secção 2.5.3). Não é necessário definir o valor Pin-Priority explicitamente no ficheiro "`/etc/apt/preferences`" apenas para usar o arquivo `experimental` a menos que deseje seguir pacotes particulares nele automaticamente para a próxima atualização.

2.7.11 Downgrade de emergência



Atenção
A utilização da técnica **apt-pinning** por um utilizador novato irá certamente causar grandes problemas. Deve evitar utilizar esta técnica exceto quando for absolutamente necessário. Veja Secção 2.1.11 para soluções alternativas.



Cuidado
O downgrade (regressão de versão) não é suportado oficialmente pelo sistema Debian por design. Deverá ser feito apenas como parte de um processo de recuperação de emergência. Apesar desta situação, é conhecido por funcionar bem em muitos incidentes. Para sistemas críticos, Deve fazer salvaguardas (backups) de todos os dados importantes após a operação de recuperação e reinstalar um sistema novo a partir da estaca zero.

Pode ter sorte ao fazer o downgrade de uma arquivo recente para um arquivo mais antigo para recuperar de uma atualização ao sistema que o deixou danificado ao manipular a **versão candidata** (veja Secção 2.7.7). Esta é uma alternativa preguiçosa às acções tediosas de muitos comandos "`dpkg -i pacote-danificado_versão-antiga.deb`" (veja Secção 2.6.3).

Procure as linhas no ficheiro "`/etc/apt/sources.list`" que acompanham `unstable` como a seguir.

```
deb http://deb.debian.org/debian/ sid main contrib non-free
```

Substitua-as de modo a acompanharem `testing`.

```
deb http://deb.debian.org/debian/ forky main contrib non-free
```

Configure o ficheiro "`/etc/apt/preferences`" como o seguinte:

```
Package: *
Pin: release a=testing
Pin-Priority: 1010
```

Corra "`apt-get update; apt-get dist-upgrade`" para forçar a regressão dos pacotes no sistema.

Remova este ficheiro especial "`/etc/apt/preferences`" após este downgrade de emergência.

Dica
É uma boa ideia remover (não purgar) o máximo de pacotes para minimizar problemas de dependências. Pode necessitar remover e instalar manualmente alguns pacotes para conseguir o downgrade do sistema. O kernel Linux, gestor de arranque, udev, PAM, APT, os pacotes relacionados com a rede e os seus ficheiros de configuração requerem atenção especial.

2.7.12 O pacote equivs

Se vai compilar um programa a partir do código-fonte para substituir um pacote Debian, o melhor é torná-lo num pacote local realmente 'debianizado' (`*.deb`) e utilizar um arquivo privado.

Se escolher compilar um programa de fonte e instalá-lo sob "`/usr/local`", pode necessitar de utilizar o `equivs` como último recurso para satisfazer as dependências em falta para o pacote.

```
Package: equivs
Priority: optional
Section: admin
Description: Circumventing Debian package dependencies
 This package provides a tool to create trivial Debian packages.
 Typically these packages contain only dependency information, but they
 can also include normal installed files like other packages do.
 .
 One use for this is to create a metapackage: a package whose sole
 purpose is to declare dependencies and conflicts on other packages so
 that these will be automatically installed, upgraded, or removed.
 .
 Another use is to circumvent dependency checking: by letting dpkg
 think a particular package name and version is installed when it
 isn't, you can work around bugs in other packages' dependencies.
 (Please do still file such bugs, though.)
```

2.7.13 Portar um pacote ao sistema stable



Cuidado
Não existe qualquer garantia de que o procedimento aqui descrito funcione sem esforços manuais adicionais devido a diferenças de sistema.

Para atualizações parciais do sistema `stable`, é desejável reconstruir um pacote dentro do ambiente dele a utilizar um pacote de código-fonte. Isto evita atualizações maciças de pacotes devido às suas dependências.

Adicione as seguintes entradas ao "`/etc/apt/sources.list`" num sistema `stable`.

```
deb-src http://deb.debian.org/debian unstable  main contrib non-free
```

Instale os pacotes necessários para a compilação e descarregue o pacote de código-fonte conforme o seguinte:

```
# apt-get update
# apt-get dist-upgrade
# apt-get install fakeroot devscripts build-essential
# apt-get build-dep foo
$ apt-get source foo
$ cd foo*
```

atualize alguns pacotes de correntes de ferramentas como o `dpkg` e o `debhelper` a partir de pacotes de backport se forem necessários para o "backporting".

Execute o seguinte.

```
$ dch -i
```

Aumentar a versão do pacote, p.e. um com "`+bp1`" acrescentado em "`debian/changelog`"

Compile os pacotes e instale-os para o sistema com o seguinte:

```
$ debuild
$ cd ..
# debi foo*.changes
```

2.7.14 Servidor proxy para o APT

Como pôr em mirror uma sub-secção inteira do arquivo Debian é um desperdício de espaço de disco e largura de banda de rede, a implantação se um servidor proxy local para o APT é desejável a ter em consideração se administrar muitos sistemas em LAN. O APT pode ser configurado para utilizar servidores proxy web genéricos (http) como o `squid` (veja Secção 6.5) conforme descrito em apt.conf(5) e em "`/usr/share/doc/apt/examples/configure-index`
A variável de ambiente "`$http_proxy`" pode ser utilizada para sobrepor a definição de servidor proxy do ficheiro "`/etc/apt/apt.conf`".

Existem ferramentas de proxy especiais para o arquivo Debian. Deve verificar o BTS antes de as utilizar.

pacote	popcon	tamanho	descrição
`apt-cacher`	V:0.28, I:0.33	265	Proxy de cache para pacotes Debian e ficheiros de código-fonte (programa Perl)
`apt-cacher-ng`	V:3.9, I:4.1	2063	Proxy de cache para distribuição de pacotes de software (programa C++ compilado)

Tabela 2.19: Lista de ferramentas proxy especiais para arquivos Debian



Cuidado
Quando Debian reorganiza a estrutura do arquivo dele, estas ferramentas de proxy especializadas tendem a necessitar ser reescritas de pelo responsável do pacote e podem não estar funcionais durante algum tempo. Por outro lado, os servidores proxy web (http) genéricos são mais robustos e mais fáceis de acompanhar estas mudanças.

2.7.15 Mais leituras sobre a gestão de pacotes

Pode aprender mais sobre a gestão de pacotes a partir das seguintes documentações.

- Documentações principais sobre a gestão de pacotes:
  - aptitude(8), dpkg(1), tasksel(8), apt(8), apt-get(8), apt-config(8), apt-secure(8), sources.list(5), apt.conf(5), e apt_preferences(5);
  - "`/usr/share/doc/apt-doc/guide.html/index.html`" e "`/usr/share/doc/apt-doc/offline.html/inde`" do pacote `apt-doc`; e
  - "`/usr/share/doc/aptitude/html/en/index.html`" do pacote `aptitude-doc-en`.
- Documentações oficiais e detalhadas no arquivo Debian:
  - "Manual de Política Debian Capítulo 2 - O Arquivo Debian",
  - "Referência dos Programadores de Debian, Capítulo 4 - Recursos para Programadores de Debian 4.6 O Arquivo Debian" e
  - "A FAQ de Debian GNU/Linux, Capítulo 6 - Os arquivos FTP de Debian".
- Tutorial para construir um pacote Debian para utilizadores de Debian:
  - "Guia para Mantedores de Debian".

Capítulo 3

A inicialização do sistema

É inteligente para si como o administrador do sistema ter uma ideia como o sistema Debian é arranca e é configurado. Apesar dos detalhes exactos estarem nos ficheiros de código-fonte dos pacotes instalados e nas suas documentações, é um pouco exagerado para a maioria de nós.

Aqui está uma visão geral aproximada dos pontos chave da inicialização do sistema Debian. Dado que o sistema Debian é um alvo em movimento, deve consultar a documentação mais recente.

- Manual do Kernel Linux Debian é a fonte primária de informação sobre o kernel Debian.
- bootup(7) descreve o processo de arranque do sistema baseado no `systemd` . (Debian Recente)
- boot(7) descreve o processo de arranque do sistema baseado em UNIX System V Release 4. (Debian Antiga)

3.1 Uma visão geral do processo de arranque

O sistema do computador passa por várias fases de processos de arranque desde o ligar da energia até que oferece, ao utilizador, o sistema operativo (SO) totalmente funcional.

Para simplicidade, limito a discussão à plataforma PC típico com a instalação predefinida.

O processo típico de arranque é como um foguete de quatro etapas. Cada etapa do foguete entrega o controle do sistema à próxima etapa.

- Secção 3.1.1
- Secção 3.1.2
- Secção 3.1.3
- Secção 3.1.4

É claro que, estes podem ser configurados de modo diferente. Por exemplo, se compilou o seu próprio kernel, pode estar a saltar o passo com o mini sistema Debian. Portanto por favor não assuma que é este o caso para o seu sistema até que o verifique por si próprio.

3.1.1 Estágio 1: a UEFI

A Unified Extensible Firmware Interface (UEFI) define um gestor de arranque como parte da especificação UEFI. Quando um computador é ligado, o gestor de arranque é a primeira fase do processo de arranque que verifica a configuração de arranque e, com base nas suas definições, executa o gestor de arranque do SO ou o kernel do

sistema operativo especificado (normalmente o gestor de arranque). A configuração de arranque é definida por variáveis armazenadas na NVRAM, incluindo variáveis que indicam os caminhos do sistema de ficheiros para os carregadores ou kernels do SO.

Uma partição de sistema EFI (ESP) é uma partição de dispositivo de armazenamento de dados que é utilizada em computadores que aderem à especificação UEFI. Acedida pelo firmware UEFI quando um computador é ligado, armazena as aplicações UEFI e os ficheiros de que estas aplicações necessitam para funcionar, incluindo os gestores de arranque do sistema operativo. (No sistema legado do PC, pode ser usada a BIOS armazenada na MBR.)

3.1.2 Estágio 2: o gestor de arranque

O gestor de arranque é o 2º estágio do processo de arranque que é iniciado pela UEFI. Ele carrega a imagem de kernel do sistema e a imagem initrd na memória e passa-lhes o controle. Esta imagem initrd é a imagem do sistema de ficheiros raiz e o suporte dele depende do gestor de arranque utilizado.

O sistema Debian normalmente usa o kernel Linux como o kernel predefinido do sistema. A imagem initrd para o atual kernel Linux 5.x é tecnicamente a imagem initramfs (initial RAM filesystem).

Existem muitos gestores de arranque e opções de configuração disponíveis.

pacote	popcon	tamanho	initrd	gestor de arranque	descrição
grub-efi-amd64	I:466	142	Suportado	GRUB UEFI	É suficientemente inteligente para compreender partições de disco e sistemas de ficheiros como vfat, ext4, ... (UEFI)
grub-pc	V:16, I:512	479	Suportado	GRUB 2	É suficientemente inteligente para compreender partições de disco e sistemas de ficheiros como vfat, ext4, ... (BIOS)
grub-rescue-pc	V:0.05, I:0.58	7269	Suportado	GRUB 2	Isto são imagens de arranque de recuperação do GRUB 2 (CD ou disquete) (Versão PC/BIOS)
grml-rescueboot	V:0.1, I:1.0	39	N/D	plug-in do GRUB 2	grml-rescueboot adiciona imagens ISO de Linux Live ao menu de arranque do grub2
syslinux	V:3, I:31	325	Suportado	Isolinux	Isto compreende o sistema de ficheiros ISO9660. É utilizado pelo CD de arranque.
syslinux	V:3, I:31	325	Suportado	Syslinux	Isto compreende o sistema de ficheiros MSDOS (FAT). É utilizado pela disquete de arranque.
loadlin	V:0.03, I:0.39	87	Suportado	Loadlin	Novo sistema é iniciado a partir do sistema FreeDOS/MSDOS.
mbr	V:0.2, I:2.7	47	Não suportado	MBR por Neil Turton	Isto é software livre que substitui o MBR do MSDOS. Apenas compreende partições de disco.

Tabela 3.1: Lista de gestores de arranque

Para o sistema UEFI, o GRUB2 lê primeiro a partição ESP e utiliza o UUID especificado para `search.fs_uuid` em "`/boot/efi/EFI/debian/grub.cfg`" para determinar a partição do ficheiro de configuração do menu GRUB2 "`/boot/grub/grub.cfg`".

A parte principal do ficheiro de configuração do menu do GRUB2 tem o seguinte aspeto:

```
menuentry 'Debian GNU/Linux' ... {
```

```
        load_video
        insmod gzio
        insmod part_gpt
        insmod ext2
        search --no-floppy --fs-uuid --set=root fe3e1db5-6454-46d6-a14c-071208ebe4b1
        echo    'Loading Linux 5.10.0-6-amd64 ...'
        linux   /boot/vmlinuz-5.10.0-6-amd64 root=UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1 ↩
            ro quiet
        echo    'Loading initial ramdisk ...'
        initrd  /boot/initrd.img-5.10.0-6-amd64
}
```

Para esta parte de `/boot/grub/grub.cfg`, esta entrada de menu significa o seguinte.

configuração	**valor**
Módulos do GRUB2 carregados	`gzio`, `part_gpt`, `ext2`
utilizada a partição raiz do sistema de ficheiros	partição identificada por `UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1`
caminho da imagem do kernel no sistema de ficheiros raiz	`/boot/vmlinuz-5.10.0-6-amd64`
utilizado parâmetro de arranque do kernel	”`root=UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1 ro quiet`”
caminho da imagem initrd no sistema de ficheiros raiz	`/boot/initrd.img-5.10.0-6-amd64`

Tabela 3.2: O significado da entrada de menu da parte acima de `/boot/grub/grub.cfg`

No sistema Debian, ”`/boot/grub/grub.cfg`” é gerido pelo pacote GRUB instalado (ex. `grub-efi-amd64`) e a modificação direta do utilizador a este ficheiro está descontinuada. Em vez disto, você deve personalizar os ficheiros de configuração em ”`/etc/grub.d/`” e ”`/etc/default/grub`”. Depois configure os ficheiros de configuração do GRUB e atualize as variáveis NVRAM para arrancar automaticamente em Debian por:

```
# dpkg-reconfigure grub-efi-amd64
```

Dica
Você pode mostrar as mensagens de registo de arranque do kernel removendo ”`quiet`” do valor ”`GRUB_CMDLINE_LINUX_DEFAULT`” em ”`/etc/default/grub`”.

Dica
Você pode adicionar um fundo no GRUB ao colocar o seu ficheiro de imagem em ”`/boot/grub/`”.

Veja ”`info grub`”, grub-install(8), grub-mkconfig(8).

3.1.3 Estágio 3: o mini-sistema Debian

O mini-sistema Debian é o 3º estágio do processo de arranque que é iniciado pelo gestor de arranque. Corre o kernel do sistema com o sistema de ficheiros raiz dele na memória. Este é um estágio preparatório opcional do processo de arranque.

Nota
O termo ”mini-sistema Debian” é cunhado pelo autor para descrever este 3º estágio do processo de arranque para este documento. Este sistema é geralmente referido como o initrd ou sistema initramfs. É utilizado pelo Instalador de Debian um sistema semelhante em memória .

O programa "`/init`" é executado como o primeiro programa neste sistema de ficheiros raiz em memória. É um programa que inicializa o kernel no espaço de utilizador e entrega o controle ao próximo estágio. Este mini-sistema Debian oferece flexibilidade ao processo de arranque tal como adicionar módulos de kernel antes do processo de arranque principal ou montar o sistema de ficheiros raiz como um encriptado.

- O programa "`/init`" é um programa de script de shell se a initramfs for criada pelo `initramfs-tools`.
  - Pode interromper esta parte do processo de arranque para obter a shell de root ao fornecer "`break=init`" etc. ao parâmetro de arranque do kernel. Veja o script "`/init`" para mais condições de interrupção. Este ambiente shell é suficientemente sofisticado para fazer uma boa inspecção do hardware da sua máquina.
  - Os comandos disponíveis neste mini-sistema Debian são versões reduzidas e disponibilizados principalmente por uma ferramenta GNU chamada busybox(1).
- O programa "`/init`" é um programa binário do `systemd` se a initramfs for criada pelo `dracut`.
  - Os comandos disponíveis neste mini-sistema Debian são versões reduzidas do ambiente systemd(1).



Cuidado
Precisa de utilizar a opção "`-n`" para o comando `mount` quando está no sistema de ficheiros raiz apenas de leitura.

3.1.4 Estágio 4: o sistema Debian normal

O sistema Debian normal é o 4º estágio do processo de arranque que é iniciado pelo mini-sistema Debian. O kernel do sistema para o mini-sistema Debian continua a correr nesse ambiente. O sistema de ficheiros raiz é mudado daquele na memória para o que está no sistema de ficheiros do dispositivo de armazenamento físico.

O programa init é executado como o primeiro programa com PID=1 para executar o processo de arranque principal de arrancar muitos programas. O caminho de ficheiro predefinido ao programa init é "`/usr/sbin/init`" mas pode ser alterado pelo parâmetro de arranque do kernel como "`init=/path/to/init_program`".

"`/usr/sbin/init`" é uma ligação simbólica para "`/lib/systemd/systemd`" após Debian 8 Jessie (Lançada em 2015).

Dica
O comando de iniciação atual do seu sistema pode ser verificado pelo comando "`ps --pid 1 -f`".

Dica
Veja Debian wiki: BootProcessSpeedup para as dicas mais recentes em como acelerar o processo de arranque.

3.2 O sistema de salvamento



Atenção
Não execute tarefas de administração do sistema sobre o processo de arranque sem ter um sistema de recuperação ou salvamento.

pacote	popcon	tamanho	descrição
systemd	V:920, I:980	11386	daemon de eventos baseado em init(8) para a concorrência (alternativa a sysvinit)
cloud-init	V:4.1, I:6.3	3267	sistema de inicialização de instâncias de infra-estruturas na nuvem
systemd-sysv	V:915, I:983	76	os manuais e ligações necessárias pelo systemd para substituir o sysvinit
init-system-helpers	V:723, I:988	132	ferramentas de ajuda para mudar entre sysvinit e systemd
initscripts	V:15, I:54	197	scripts para inicializar e desligar o sistema
sysvinit-core	V:3.1, I:3.6	365	utilitários de init(8) estilo System-V
sysv-rc	V:29, I:57	85	mecanismo de mudança de runlevel estilo System-V
sysvinit-utils	V:725, I:1000	100	utilitários estilo System-V (startpar(8), bootlogd(8), …)
lsb-base	V:205, I:300	12	Linux Standard Base funcionalidade de script de init 3.2
insserv	V:31, I:57	132	ferramenta para organizar a sequência de arranque a usar dependências dos scripts init.d LSB
kexec-tools	V:1.4, I:5.3	320	ferramenta kexec para re-arranques kexec(8) (re-arranque a quente)
systemd-bootchart	V:0.17, I:0.52	131	analisador de performance do processo de arranque
mingetty	V:0.2, I:2.3	36	getty(8) apenas de consola
mgetty	V:0.13, I:0.47	318	substituto inteligente de modem getty(8)

Tabela 3.3: Lista de utilitários de arranque para o sistema Debian

A disponibilidade dum sistema de salvamento permite-nos executar tarefas desafiantes tais como:

- Arrancar um sistema a partir de uma instalação de carregador de arranque quebrada
- Corrigir uma instalação de carregador de arranque quebrada
- Extrair dados de um sistema quebrado que não arranca
- Editar sistemas de ficheiros, partições de disco, e volumes LVM envolvendo o sistema de ficheiros raiz

Tipicamente, um sistema de salvamento é fornecido como um ficheiro de imagem ISO escrito numa média de armazenamento amovível tal como:

- Memória flash USB preparada como Secção 9.7.2
- CD / DVD preparado como Secção 9.7.7

Para simplicidade, os casos com dispositivos flash USB são mencionados como exemplos, mas CD ou DVD também pode ser usado.

Dica
Você pode precisar de alterar algumas variáveis UEFI NVRAM para arrancar carregadores de arranque arbitrários na média de armazenamento amovível.

3.2.1 Sistema de salvamento GRUB UEFI em USB

O sistema de salvamento GRUB UEFI com menu pode ser arrancado ao ligar a energia do sistema com o ”Sistema de salvamento GRUB UEFI em USB” inserido.

Este ”Sistema de salvamento GRUB UEFI em USB” é preparado a escrever a imagem ISO Super Grub2 Disk numa Pen flash USB antecipadamente.

No caso de configuração de carregador de arranque quebrada pela instalação de outro sistema operativo etc., você pode corrigir isto por:

- Arranque o sistema de salvamento GRUB UEFI para descobrir automaticamente sistemas instalados com arranque.
- Arranque o sistema Debian instalado a partir do menu do GRUB.
- Na linha de comandos shell do root em Linux:

```
# dpkg-reconfigure grub-efi-amd64
```

Nota
A imagem ISO de arranque de salvamento do GRUB também pode ser gerada seguindo ”`info grub-mkrescue`”. Oferece uma linha de comandos shell CLI GRUB mas não oferece descoberta automática de sistemas instalados com arranque.

3.2.2 Sistema de salvamento Linux live em USB

O sistema de salvamento Linux live pode ser arrancado ao ligar a energia do sistema com o ”Sistema de salvamento Linux live em USB” inserido.

Este ”Sistema de salvamento Linux live em USB” pode ser preparado a escrever uma das imagens ISO Linux live baseadas em Debian Pen flash USB antecipadamente. Aqui estão alguns exemplos de tais imagens Linux live.

- Imagens Live de Debian
- Kali Linux Live
- Grml Live Linux

Aqui estão alguns casos de utilização deste sistema de salvamento Linux live:

- Corrigir a configuração de carregador de arranque danificada causado pela instalação de outro sistema operativo, etc.:
  - Arranque o sistema de salvamento Linux live.
  - Monte a partição que contém o sistema de ficheiros raiz do sistema Debian instalado que não arranca em ”`/mnt`”.
  - Na linha de comandos shell do root em Linux:

```
# chroot /mnt; dpkg-reconfigure grub-efi-amd64
```

- Corrigir o pacote `dpkg` quebrado:
  - Arranque o sistema de salvamento Linux live.
  - Monte a partição que contém o sistema de ficheiros raiz do sistema Debian instalado com o pacote `dpkg` quebrado em ”`/mnt`”.
  - Na linha de comandos shell do root em Linux:

```
# dpkg --root /mnt -i /mnt/var/cache/apt/archives/dpkg_old_version_amd64.deb
```

- Executar alterações normalmente proibidas tais como operações no sistema de ficheiros no sistema instalado (veja Secção 9.6).



Atenção
O sue ecrã GUI do sistema Linux live pode ficar trancado após inatividade.

Dica

- É uma boa ideia definir a sua palavra passe assim que arranca um sistema Linux live.
- Você pode definir a sua palavra passe fazendo ”`sudo passwd` *`utilizador`*” a partir do aviso de shell do utilizador nas consolas virtuais de Linux (veja Secção 1.1.6).
- Alguns sistemas Linux live podem definir os seus ”*utilizador*/*palavra passe*” predefinidos: ”Debian Live” = ”user/live”, ”Kali Linux Live” = ”kali/kali”

3.2.3 Sistema de salvamento Linux live a partir do GRUB

O sistema de salvamento Linux live pode ser arrancado a partir duma entrada de menu do GRUB. A configuração do GRUB para isto é preparada pelo seguinte:

- Instale o pacote `grml-rescueboot`
- Encontre uma imagem ISO Linux live que tenha ”`/boot/grub/loopback.cfg`” na sua imagem.
  - As imagens ISO mencionadas em Secção 3.2.2 têm ”`/boot/grub/loopback.cfg`” na sua imagem.
- Copie algumas destas imagens ISO Linux live para o directório ”`/boot/grml/`”.
- Atualize o menu do GRUB com:

```
# dpkg-reconfigure grub-efi-amd64
```

Após ligar a energia do sistema, o GRUB mostra um menu com candidatos para sistemas de salvamento Linux live.

3.3 Systemd

3.3.1 init do Systemd

Quando o sistema Debian arranca, `/usr/sbin/init` com uma ligação simbólica a `/usr/lib/systemd` é iniciado como o processo do sistema init (`PID=1`) cujo dono é o root (`UID=0`). Veja systemd(1).

O processo init do `systemd` espalha processos em paralelo com base nos arquivos de configuração do unit (veja systemd.unit(5)) os quais são escritos em estilo declarativo em vez do estilo processual tipo SysV.

Os processos gerados são colocados em grupos de controlo individuais do Linux com o nome da unidade a que pertencem na hierarquia privada do systemd (ver cgroups e Secção 4.7.5).

As unidades para o modo de sistema são carregadas a partir do ”Caminho de pesquisa de unidade de sistema” descrito em systemd.unit(5). As principais são as seguintes, por ordem de prioridade:

- ”`/etc/systemd/system/*`”: Unidades de sistema criadas pelo administrador
- ”`/run/systemd/system/*`”: Unidades de tempo de execução

- "`/lib/systemd/system/*`": Unidades de sistema instaladas pelo gestor de pacotes da distribuição

As suas inter-dependências são especificadas pelas directivas "`Wants=`", "`Requires=`", "`Before=`", "`After=`", ... (veja "MAPPING OF UNIT PROPERTIES TO THEIR INVERSES" em systemd.unit(5)). Os controlos de recursos estão também definidos (veja systemd.resource-control(5)).

O sufixo do ficheiro de configuração da unidade codifica os seus tipos como:

- ***.service** descreve o processo controlado e supervisionado pelo `systemd`. Veja systemd.service(5).
- ***.device** descreve o dispositivo exposto em sysfs(5) como uma árvore de dispositivos do udev(7). Veja systemd.device(5).
- ***.mount** descreve o ponto de montagem do sistema de ficheiros controlado e supervisionado pelo `systemd`. Veja systemd.mount(5).
- ***.automount** Descreve o ponto de montagem automático do sistema de ficheiros controlado e supervisionado pelo `systemd`. Veja systemd.automount(5).
- ***.swap** descreve o dispositivo ou ficheiro de memória virtual (swap) controlado e supervisionado pelo `systemd`. Veja systemd.swap(5).
- ***.path** descreve o caminho monitorizado pelo `systemd` para activação baseada-no-caminho. Veja systemd.path(5).
- ***.socket** descreve o socket controlado e supervisionado pelo `systemd` para activação baseada-em-socket. Veja systemd.socket(5).
- ***.timer** descreve o temporizador controlado e supervisionado pelo `systemd` para activação baseada-em-temporização. Veja systemd.timer(5).
- ***.slice** gere recursos com cgroups(7). Veja systemd.slice(5).
- ***.scope** é criado programaticamente a usar as interfaces de barramento do `systemd` para gerir um conjunto de processos do sistema. Veja systemd.scope(5).
- ***.target** agrupa outros ficheiros de configuração de unit para criar o ponto de sincronização durante o arranque. Veja systemd.target(5).

Após o arranque do sistema (o, init), o processo `systemd` tenta arrancar o "`/lib/systemd/system/default.target` (que normalmente é uma ligaö#ao simbólica para "`graphical.target`"). Primeiro, algumas unidades alvo especiais (veja systemd.special(7)) tais como "`local-fs.target`", "`swap.target`" e "`cryptsetup.target`" são puxadas para montar os sistemas de ficheiros. Depois, outras unidades alvo são também puxadas pelas dependências da unidade alvo. Para mais detalhes. leia bootup(7).

O `systemd` oferece funcionalidades de compatibilidade regressiva. Os scripts de arranque estilo SysV em "`/etc/init.d/r` são ainda analisados e telinit(8) é traduzido em pedidos activação de unidade do systemd.



Cuidado
Os runlevel 2 a 4 emulados são todos direccionados por uma ligação simbólica ao mesmo "`alvo de multi-utilizador`".

3.3.2 Inicio de sessão Systemd

Quando um utilizador se liga ao sistema Debian via gdm3(8), sshd(8), etc., `/lib/systemd/system --user` é iniciado como o processo gestor de serviços de utilizador pertencente ao utilizador correspondente. Veja systemd(1).

O processo do gestor de serviços do utilizador `systemd` gera processos em paralelo com base nos ficheiros de configuração da unidade declarativa (ver systemd.unit(5) e `user@.service`(5)).

As unidades para o modo de utilizador são carregadas a partir do "Caminho de pesquisa de unidade de sistema" descrito em systemd.unit(5). As principais são as seguintes, por ordem de prioridade:

- "`~/.config/systemd/user/*`": Unidades de configuração do utilizador
- "`/etc/systemd/user/*`": Unidades de utilizador criadas pelo administrador
- "`/run/systemd/user/*`": Unidades de tempo de execução
- "`/lib/systemd/user/*`": Unidades de utilizador instaladas pelo gestor de pacotes de distribuição

Estes são geridos da mesma forma que os Secção 3.3.1.

3.4 A mensagem do kernel

As mensagens de erros do kernel mostradas na consola podem ser configuradas ao definir o nível de limiar dele.

```
# dmesg -n3
```

valor de nível de erro	nome de nível de erro	significado
0	KERN_EMERG	sistema está inutilizável
1	KERN_ALERT	acção tem de ser tomada imediatamente
2	KERN_CRIT	condições críticas
3	KERN_ERR	condições de erro
4	KERN_WARNING	condições de aviso
5	KERN_NOTICE	condição normal mas significante
6	KERN_INFO	informativa
7	KERN_DEBUG	mensagens de nível de depuração

Tabela 3.4: Lista de níveis de erro do kernel

3.5 A mensagem do sistema

Sob `systemd`, ambas mensagens de kernel e sistema são registadas pelo serviço journal `systemd-journald.service` (também conhecido por `journald`) seja em dados binários persistentes em "`/var/log/journal`" ou em dados binários voláteis em "`/run/log/journal/`". Estes dados de relatório binários são acedidos pelo comando journalctl(1). Por exemplo, pode apresentar o registo do último arranque como:

```
$ journalctl -b
```

Operação	Fragmentos de comando
Ver o registo dos serviços do sistema e do kernel desde o último arranque	"`journalctl -b --system`"
Ver o registo dos serviços do utilizador atual desde o último arranque	"`journalctl -b --user`"
Ver o registo de tarefas de" `$unit`" do último arranque	"`journalctl -b -u $unit`"
Ver o registo do trabalho de "`$unit`" (estilo "`tail -f`") desde o último arranque	"`journalctl -b -u $unit -f`"

Tabela 3.5: Lista de trechos típicos do comando `journalctl`

Sob o `systemd`, o utilitário de registo do sistema rsyslogd(8) pode ser desinstalado. Se for instalado, altera o seu comportamento para ler os dados de registo binários voláteis (em vez do padrão pré-systemd "`/dev/log`") e para criar dados de registo do sistema ASCII permanentes tradicionais. Isto pode ser personalizado por "`/etc/default/rsyslog`" e "`/etc/rsyslog.conf`" tanto para o ficheiro de registos como para a apresentação no ecrã. Veja rsyslogd(8) e rsyslog.conf(5). Veja também Secção 9.3.2.

3.6 Gestão do sistema

O `systemd` oferece não apenas o sistema init, mas também operações genéricas de gestão de sistema com o comando systemctl(1).

Aqui, "`$unit`" nos exemplos em cima pode ser um único nome de unidade (sufixos como `.service` e `.target` são opcionais) ou, em muitos casos, especificações de múltiplas unidades (a simbologia da shell "`*`", "?", "`[]`" a utilizar fnmatch(3) serão correspondidos aos nomes primários de todas as unidades presentemente em memória).

Os comandos de alteração do estado do sistema nos exemplos em cima são tipicamente precedidos por "`sudo`" para obter os privilégios administrativos necessários.

Os resultados de "`systemctl status $unit|$PID|$dispositivo`" usam cores no ponto ("●") para sumarizar rapidamente o estado da unidade.

- Ponto "●" branco indica estado "inativo" ou "desactivado".
- Ponto "●" vermelho indica um estado de "falha" ou "erro".
- Ponto "●" verde indica um estado "ativo", "a reiniciar" ou "a ativar".

3.7 Outros monitores de sistema

Aqui está uma lista de outros trechos de comandos de monitorização sob o `systemd`. Por favor, leia as páginas de manual pertinentes, incluindo cgroups(7).

3.8 Configuração do sistema

3.8.1 O nome da máquina

O kernel mantém o **nome-de-máquina** do sistema. A unidade do sistema arrancado por `systemd-hostnamed.service` define o nome de máquina do sistema durante o arranque ao nome armazenado em "`/etc/hostname`". Este ficheiro deve conter **apenas** o nome de máquina do sistema e não um nome de domínio totalmente qualificado.

Para escrever o nome de máquina atual corra hostname(1) sem argumentos.

3.8.2 O sistema de ficheiros

As opções de montagem de sistemas de ficheiros de discos normais e de rede são definidas em "`/etc/fstab`". Veja fstab(5) e Secção 9.6.7.

A configuração do sistema de ficheiros encriptado é definida em "`/etc/crypttab`". Veja crypttab(5)

A configuração do software RAID com mdadm(8) é definida em "`/etc/mdadm/mdadm.conf`". Veja mdadm.conf(5).



Atenção
Após montar todos os sistemas de ficheiros, os ficheiros temporários em "`/tmp`", "`/var/lock`" e "`/var/run`" são limpos para cada arranque.

Operação	Fragmentos de comando
Lista todos os tipos de unidades disponíveis	"systemctl list-units --type=help"
Lista todas as unidades alvo na memória	"systemctl list-units --type=target"
Lista todas as unidades de serviço na memória	"systemctl list-units --type=service"
Lista todas as unidades de dispositivo na memória	"systemctl list-units --type=device"
Lista todas as unidades de montagem na memória	"systemctl list-units --type=mount"
Lista todas unidades de socket em memória	"systemctl list-sockets"
Lista todas as unidades de temporizador em memória	"systemctl list-timers"
Iniciar o "$unit"	"systemctl start $unit"
Parar o "$unit"	"systemctl stop $unit"
Recarregar configuração específica do serviço	"systemctl reload $unit"
Parar e iniciar todo "$unit"	"systemctl restart $unit"
Iniciar o "$unit" e parar todos os outros	"systemctl isolate $unit"
Mudar para "gráfico" (sistema GUI)	"systemctl isolate graphical"
Mudar para "multi-utilizador" (sistema CLI)	"systemctl isolate multi-user"
Mudar para "recuperação" (sistema CLI de único utilizador)	"systemctl isolate rescue"
Enviar sinal kill ao "$unit"	"systemctl kill $unit"
Verificar se o serviço "$unit" está ativo	"systemctl is-active $unit"
Verificar se o serviço "$unit" falhou	"systemctl is-failed $unit"
Verifica o estado de "$unit\|$PID|dispositivo"	"systemctl status $unit\|$PID|$device"
Mostra propriedades de 1"$unit\|$job"	"systemctl show $unit\|$job"
Reinicia um "$unit" falhado	"systemctl reset-failed $unit"
List dependências de todos os serviços unit	"systemctl list-dependencies --all"
Lista ficheiros unit instalados no sistema	"systemctl list-unit-files"
Ativa "$unit" (adicionar ligação simbólica)	"systemctl enable $unit"
Desactiva "$unit" (remove ligação simbólica)	"systemctl disable $unit"
Desmascara "$unit" (remove ligação simbólica para "/dev/null")	"systemctl unmask $unit"
Mascara "$unit" (adicionar ligação simbólica para "/dev/null")	"systemctl mask $unit"
Obter definição de alvo-predefinido	"systemctl get-default"
Define alvo-predefinido para "graphical" (sistema GUI)	"systemctl set-default graphical"
Define alvo-predefinido para "multi-user" (sistema CLI)	"systemctl set-default multi-user"
Mostra ambiente da função	"systemctl show-environment"
Define "variável" de ambiente de função para "valor"	"systemctl set-environment variable=value"
Remove a definição da "variável" de ambiente de função	"systemctl unset-environment variable"
Reinicia todos os ficheiros unit e os daemons	"systemctl daemon-reload"
Desligar o sistema	"systemctl poweroff"
Desligar e reiniciar o sistema	"systemctl reboot"
Suspender o sistema	"systemctl suspend"
Hibernar o sistema	"systemctl hibernate"

Tabela 3.6: Lista de trechos de comandos típicos de gestão do systemctl

Operação	Fragmentos de comando
Mostra o tempo gasto em cada passo de inicialização	"`systemd-analyze time`"
Lista de todas as unidades pelo tempo de inicialização	"`systemd-analyze blame`"
Carrega e detecta erros no ficheiro "`$unit`"	"`systemd-analyze verify $unit`"
Apresentar informações breves sobre o estado do tempo de execução do utilizador que efectua a chamada	"`loginctl user-status`"
Mostrar informação concisa sobre o estado do tempo de execução da sessão do utilizador	"`loginctl session-status`"
Acompanhar o processo de arranque através dos cgroups	"`systemd-cgls`"
Acompanhar o processo de arranque através dos cgroups	"`ps xawf -eo pid,user,cgroup,args`"
Acompanhar o processo de arranque através dos cgroups	Lê sysfs sob "`/sys/fs/cgroup/`"

Tabela 3.7: Lista de outros trechos de comandos de monitorização `systemd`

3.8.3 Inicialização da interface de rede

As interfaces de rede são tipicamente iniciadas em "`networking.service`" para a interface `lo` e "`NetworkManager.ser` para as outras interfaces em sistemas Debian modernos sob `systemd`.

Veja Capítulo 5 para saber como os configurar.

3.8.4 Inicialização do sistema de nuvem

A instância do sistema de nuvem pode ser lançada como um clone de "Debian Official Cloud Images" ou imagens semelhantes. Para essa instância do sistema, personalidades como nome de máquina, sistema de ficheiros, rede, localidade, chaves SSH, utilizadores e grupos podem ser configurados usando funcionalidades fornecidas pelos pacotes `cloud-init` e `netplan.io` com várias fontes de dados, como ficheiros colocados na imagem original do sistema e dados externos fornecidos durante seu lançamento. Esses pacotes permitem a configuração declarativa do sistema usando dados YAML.

Veja mais em "Computação em nuvem com Debian e seus descendentes", "Documentação de iniciação na Nuvem" e Secção 5.4.

3.8.5 Exemplo de personalização para ajustar o serviço sshd

Com uma instalação predefinida, muitos serviços de rede (veja Capítulo 6) são arrancados como processos daemon após `network.target` durante o arranque do sistema pelo `systemd`. O "`sshd`" não é excepção. Vamos mudar isto para arranque a-pedido do "`sshd`" como um exemplo de personalização.

Primeiro, desativar a unidade de serviço instalada no sistema.

```
$ sudo systemctl stop sshd.service
$ sudo systemctl mask sshd.service
```

O sistema de ativação de sockets a pedido dos serviços Unix clássicos era feito através do superservidor `inetd` (ou `xinetd`). No `systemd`, o equivalente pode ser ativado adicionando ficheiros de configuração de unidades ***.socket** e ***.service**.

`sshd.socket` para especificar um socket onde escutar

```
[Unit]
Description=SSH Socket for Per-Connection Servers

[Socket]
ListenStream=22
Accept=yes

[Install]
WantedBy=sockets.target
```

`sshd@.service` como o ficheiro de serviço correspondente do `sshd.socket`

```
[Unit]
Description=SSH Per-Connection Server

[Service]
ExecStart=-/usr/sbin/sshd -i
StandardInput=socket
```

Depois reinicie.

```
$ sudo systemctl daemon-reload
```

3.9 O sistema udev

O sistema udev fornece um mecanismo para a descoberta e inicialização automática de hardware (veja udev(7)) desde o kernel Linux 2.6. Após a descoberta de cada dispositivo pelo kernel, o sistema udev inicia um processo de utilizador que usa informação do sistema de ficheiros sysfs (ver Secção 1.2.12), carrega os módulos necessários do kernel que o suportam usando o programa modprobe(8) (ver Secção 3.10), e cria os nós de dispositivos correspondentes.

Dica
Se "`/lib/modules/`*`versão-de-kernel`*`/modules.dep`" não foi gerado de modo apropriado pelo depmod(8) por alguma razão, os módulos podem não carregar como esperado pelo sistema udev. Execute "`depmod -a`" para o corrigir.
Para regras de montagem em "`/etc/fstab`", os nós de dispositivos não precisam de ser os estáticos. Pode usar o UUID para montar os dispositivos em vez dos nomes de dispositivo como "`/dev/sda`". Veja Secção 9.6.3.

Como o sistema udev é de certa forma um alvo em movimento, deixo os detalhes para outras documentações e descrevo a informação mínima aqui.



Atenção
Não tente executar programas de longa duração como o script de backup com `RUN` nas regras do udev como mencionado em udev(7). Por favor, crie um ficheiro systemd.service(5) adequado e active-o. Veja Secção 10.2.3.2.

3.10 A inicialização de módulos do kernel

O programa modprobe(8) permite-nos configurar o kernel Linux em execução a partir do processo de utilizador ao adicionar e remover módulos do kernel. O sistema udev (veja Secção 3.9) automatiza a invocação dele para ajudar na inicialização dos módulos de kernel.

Existem módulos de não-hardware e módulos driver de hardware especial como os seguintes que precisam de ser pré-carregados ao listá-los no ficheiro "`/etc/modules`" (veja modules(5)).

- módulos TUN/TAP que disponibilizam dispositivos de rede Point-to-Point virtuais (TUN) e dispositivos de rede Ethernet virtuais (TAP),
- módulos netfilter que disponibilizam capacidades de firewall netfilter (iptables(8), Secção 5.7) e
- módulos de driver watchdog timer.

Os ficheiros de configuração para o programa modprobe(8) estão localizados sob o diretório "`/etc/modprobes.d/`" como explicado em modprobe.conf(5). (Se deseja evitar que alguns módulos do kernel sejam carregados automaticamente, considere metê-los em lista negra no ficheiro "`/etc/modprobes.d/blacklist`".)

O ficheiro "`/lib/modules/`*`version`*`/modules.dep`" gerado pelo programa depmod(8) descreve as dependências dos módulos usados pelo programa modprobe(8).

Nota
Se tiver problemas com o carregamento de módulos durante o arranque ou com o modprobe(8), "`depmod -a`" pode resolver esses problemas ao reconstruir "`modules.dep`".

O programa modinfo(8) mostra informação sobre um módulo do kernel Linux.

O programa lsmod(8) formata lindamente o conteúdo de "`/proc/modules`" e mostra que módulos do kernel que estão atualmente carregados.

Dica
Pode identificar o hardware exacto no seu sistema. Veja Secção 9.5.3.
Pode configurar o hardware durante o arranque para ativar as funcionalidades esperadas do hardware. Veja Secção 9.5.4.
Pode provavelmente adicionar suporte para o seu dispositivo especial ao recompilar o kernel. Veja Secção 9.10.

Capítulo 4

Autenticação e controlos de acesso

Quando uma pessoa (ou programa) requer acesso ao sistema, a autenticação confirma a identidade para ser de confiança.



Atenção
Erros de configuração do PAM podem trancá-lo fora do seu sistema. Terá de ter um CD de recuperação à mão ou arrancar por uma partição de arranque alternativa. Para recuperar, arranque o sistema com eles e corrija a partir daí.

4.1 Autenticação normal de Unix

A autenticação normal de Unix é disponibilizada pelo módulo pam_unix(8) sob PAM (Pluggable Authentication Modules). Os seus 3 ficheiros de configuração importantes, com entradas separadas por "`:`", são os seguintes.

ficheiro	permissão	utilizador	grupo	descrição
`/etc/passwd`	`-rw-r--r--`	`root`	`root`	informação da conta do utilizador (filtrada)
`/etc/shadow`	`-rw-r-----`	`root`	`shadow`	informação segura da conta do utilizador
`/etc/group`	`-rw-r--r--`	`root`	`root`	informação do grupo

Tabela 4.1: 3 ficheiros de configuração importantes para pam_unix(8)

"`/etc/passwd`" contém o seguinte.

```
 ...
user1:x:1000:1000:User1 Name,,,:/home/user1:/bin/bash
user2:x:1001:1001:User2 Name,,,:/home/user2:/bin/bash
 ...
```

Como explicado em passwd(5), cada entrada separada por "`:`" neste ficheiro significa o seguinte.

- Nome de login
- Entrada de especificação de palavra-passe
- ID numérico do utilizador

- ID numérico do grupo
- Nome de utilizador ou campo de comentários
- Directório home do utilizador
- Interpretador de comandos opcional do utilizador

A segunda entrada de "`/etc/passwd`" foi utilizada para a entrada de palavra-passe encriptada. Após a introdução de "`/etc/shadow`", esta entrada é utilizada para a entrada de especificação da palavra-passe.

conteúdo	**significado**
(vazio)	conta sem palavra-passe
x	a palavra-passe encriptada está em "`/etc/shadow`"

Tabela 4.2: A segunda entrada no conteúdo de "`/etc/passwd`"

"`/etc/shadow`" contém o seguinte.

```
 ...
user1:$1$Xop0FYH9$IfxyQwBe9b8tiyIkt2P4F/:13262:0:99999:7:::
user2:$1$vXGZLVbS$ElyErNf/agUDsm1DehJMS/:13261:0:99999:7:::
 ...
```

Como explicado em shadow(5), cada entrada separada por "`:`" neste ficheiro significa o seguinte.

- Nome de login
- Palavra-passe encriptada (O "1" inicial indica o uso de encriptação MD5. O "*" indica nenhum login.)
- Data da última alteração da palavra-passe, expressada a quantidade de dias desde 1 de Janeiro de 1970
- Quantidade de dias que o utilizador terá que esperar antes de ser-lhe permitido alterar a palavra-passe outra vez
- Quantidade de dias após os quais o utilizador terá que alterar a palavra passe dele
- Quantidade de dias antes de uma palavra-passe expirar durante o qual o utilizador deve ser avisado
- Quantidade de dias após uma palavra-passe ter expirador durante os quais a palavra-passe deve ainda ser aceite
- Data de expiração de uma conta, expressada como a quantidade de dias desde 1 de Janeiro de 1970
- …

"`/etc/group`" contém o seguinte.

```
group1:x:20:user1,user2
```

Como explicado em grupo(5), cada entrada separada por "`:`" neste ficheiro significa o seguinte.

- Nome do grupo
- Palavra-passe encriptada (na realidade não utilizada)
- ID numérico do grupo
- lista de nomes de utilizadores, separada por ","

Nota
"`/etc/gshadow`" disponibiliza uma função semelhante à do "`/etc/shadow`" para "`/etc/group`" mas não é realmente usado.

Nota
A quantidade real de membros de um grupo de um utilizador pode ser adicionada dinamicamente se a linha "`auth optional pam_group.so`" for adicionada ao "`/etc/pam.d/common-auth`" e definida em "`/etc/security/group.conf`". Veja pam_group(8).

Nota
O pacote `base-passwd` contém uma lista autorizada do utilizador e do grupo: "`/usr/share/doc/base-passwd/users-and-groups.html`".

4.2 Gerir informação de conta e palavra-passe

Aqui estão alguns comandos notáveis para gerir informação de conta.

comando	função
`getent passwd` *`user_name`*	navegar na informação da conta de "*`nome_de_utilizador`*"
`getent shadow` *`user_name`*	explorar informação confidencial (shadow) da conta de "*`nome_de_utilizador`*"
`getent group` *`group_name`*	navegar na informação do grupo de "*`nome_do_grupo`*"
`passwd`	gerir a palavra-passe da conta
`passwd -e`	definir palavra-passe para uma vez para a activação da conta
`chage`	gerir a informação de envelhecimento da palavra-passe

Tabela 4.3: Lista de comandos para gerir informação de conta

Pode necessitar de privilégios de root para algumas funções funcionarem. Veja crypt(3) pela encriptação de palavra-passe e dados.

Nota
Num sistema configurado com PAM e NSS como a máquina salsa de Debian, o conteúdo dos "`/etc/passwd`", "`/etc/group`" e "`/etc/shadow`" locais pode não ser utilizado activamente pelo sistema. Os comandos em cima são válidos mesmo sob tal ambiente.

4.3 Boa palavra-passe

Quando criar uma conta durante a instalação do seu sistema ou com o comando passwd(1), deve escolher uma boa palavra-passe que consista pelo menos de 6 a 8 caracteres incluindo um ou mais caracteres de cada um dos seguintes conjuntos de acordo com passwd(1).

- Alfabéticos de minúsculas
- Dígitos de 0 a 9
- Marcas de pontuação



Atenção
Não escolha palavras adivinháveis como palavra-passe. O nome de conta, número de segurança social, número de telefone, endereço, data de nascimento, nomes de membros de família ou de animais, palavras do dicionário, sequências simples de caracteres como "12345" ou "qwerty", ... são todas más escolhas para a palavra-passe.

4.4 Criar palavra-passe encriptada

Existem ferramentas independentes para gerar palavras passe encriptadas com sal.

pacote	popcon	tamanho	comando	função
whois	V:23, I:203	384	mkpasswd	frontend cheio de funcionalidades para a biblioteca crypt(3)
openssl	V:846, I:996	2564	openssl passwd	computa hashes de palavras-passe (OpenSSL). passwd(1ssl)

Tabela 4.4: Lista de ferramentas para gerar palavras-passe

4.5 PAM e NSS

Os sistemas modernos tipo-Unix como o sistema Debian disponibilizam mecanismos PAM (Pluggable Authentication Modules) e NSS (Name Service Switch) para o administrador local configurar o sistema dele. O papel destes pode ser resumido ao seguinte.

- O PAM oferece um mecanismo de autenticação flexível utilizado pelo software de aplicações e assim involve trocas de dados de palavra-passe.
- O NSS oferece um mecanismo de serviço de nomes flexível que é utilizado frequentemente pela biblioteca C standard para obter o nome de utilizador e grupo para programas como o ls(1) e o id(1).

Estes sistemas PAM e NSS necessitam ser configurados de modo consistente.

Os pacotes notáveis dos sistemas PAM e NSS são os seguintes.

pacote	popcon	tamanho	descrição
libpam-modules	V:942, I:1000	892	Pluggable Authentication Modules (serviço básico)
libpam-ldapd	V:7, I:13	79	Pluggable Authentication Module que permite interfaces LDAP
libpam-systemd	V:764, I:969	848	Pluggable Authentication Module para registar sessões de utilizador para o logind
libpam-doc	I:7.5	1492	Pluggable Authentication Modules (documentação em html e texto)
libc6	V:936, I:1000	5483	GNU C Library: Bibliotecas de partilha que também disponibilizam o serviço "Name Service Switch"
glibc-doc	I:5.1	3927	Biblioteca GNU C: Manuais (Manpages)
glibc-doc-reference	I:3.1	14764	Biblioteca GNU C: Manual de referência em info, pdf e html (não-livre)
libnss-mdns	V:299, I:507	142	Módulo NSS para resolução de nomes Multicast DNS
libnss-ldapd	V:8, I:16	131	Módulo NSS para utilizar o LDAP como um serviço nomeador

Tabela 4.5: Lista de sistemas PAM e NSS notáveis

- "O Guia do Administrador do Sistema Linux-PAM" em libpam-doc é essencial para aprender configuração do PAM.
- A secção "System Databases e Name Service Switch" em glibc-doc-reference é essencial para aprender a configuração do NSS.

Nota
Pode ver uma lista mais extensa e atual pelo comando "`aptitude search 'libpam-|libnss-'`". O acrónimo NSS também pode significar "Network Security Service" o que é diferente de "Name Service Switch".

Nota
PAM é a maneira mais básica de inicializar variáveis de ambiente para cada programa com o valor predefinido de todo o sistema.

Sob systemd, o pacote `libpam-systemd` é instalado para gerir os logins de utilizador ao registar as sessões de utilizador na hierarquia do grupo de control do `systemd` para logind. veja systemd-logind(8), logind.conf(5),e pam_systemd(8).

4.5.1 Ficheiros de configuração acedidos pelo PAM e NSS

Aqui estão alguns ficheiros de configuração notáveis acedidos pelo PAM e NSS.

ficheiro de configuração	função
`/etc/pam.d/`*`program_name`*	defina a configuração do PAM para o programa "*`nome_do_programa`*"; veja pam(7) e pam.d(5)
`/etc/nsswitch.conf`	define a configuração NSS com a entrada para cada serviço. Veja nsswitch.conf(5)
`/etc/nologin`	limita o login de utilizador pelo módulo pam_nologin(8)
`/etc/securetty`	limita a tty ao acesso de root pelo módulo pam_securetty(8)
`/etc/security/access.conf`	define limites de acesso pelo módulo pam_access(8)
`/etc/security/group.conf`	define a restrição baseada em grupo pelo módulo pam_group(8)
`/etc/security/pam_env.conf`	define as variáveis de ambiente pelo módulo pam_env(8)
`/etc/environment`	define variáveis de ambiente adicionais pelo módulo pam_env(8) com o argumento "`readenv=1`"
`/etc/default/locale`	define o locale pelo módulo pam_env(8) com o argumento "`readenv=1 envfile=/etc/default/locale`" (Debian)
`/etc/security/limits.conf`	define restrição de recursos (ulimit, core, …) pelo módulo pam_linits(8)
`/etc/security/time.conf`	define a retenção de tempo pelo módulo pam_time(8)
`/etc/systemd/logind.conf`	define a configuração do gestor de login do `systemd` (veja logind.conf(5) e systemd-logind.service(8))

Tabela 4.6: Lista de ficheiros de configuração acedidos pelo PAM e NSS

A limitação da seleção da palavra-passe é implementada pelos módulos do PAM, pam_unix(8) e pam_cracklib(8). Eles podem ser configurados com os argumentos deles.

Dica
Os módulos PAM utilizam o sufixo "`.so`" para os nomes de ficheiros deles.

4.5.2 O moderno sistema de gestão centralizado

A moderna gestão de sistema centralizada pode ser implantada a utilizar o servidor centralizado Lightweight diretory Access Protocol (LDAP) para administrar muitos sistemas tipo-Unix e não-tipo-Unix na rede. A implementação de código aberto do Lightweight diretory Access Protocol é o Software OpenLDAP.

O servidor LDAP disponibiliza para o sistema Debian a informação de conta através do uso de PAM e NSS com os pacotes `libpam-ldapd` e `libnss-ldapd`. São necessárias várias acções para ativar isto (não utilizei esta configuração e o seguinte é puramente informação secundária. Por favor leia isto neste contexto.).

- Configura um servidor LDAP centralizado ao correr um programa como o daemon de LDAP autónomo slapd(8).
- Altera os ficheiros de configuração do PAM no diretório "`/etc/pam.d/`" para utilizar "`pam_ldap.so`" em vez do predefinido "`pam_unix.so`".
- Altera a configuração do NSS no ficheiro "`/etc/nsswitch.conf`" para usar "`ldap`" em vez da predefinição("`compat`" ou "`file`").
- Tem de fazer `libpam-ldapd` para usar a ligação SSL (ou TLS) para a segurança da palavra-passe.
- Pode fazer a `libnss-ldapd` usar ligação SSL (ou TLS) para assegurar a integridade dos dados à custa de maior sobrecarga da rede LDAP.
- Deve correr o nscd(8) localmente para pôr em cache quaisquer resultados de busca LDAP de modo a reduzir o tráfego de rede do LDAP.

Veja as documentações em nsswitch.conf(5), pam.conf(5), ldap.conf(5), e "`/usr/share/doc/libpam-doc/html/`" oferecidas pelo pacote `libpam-doc` e "`info libc 'Name Service Switch'`" oferecida pelo pacote `glibc-doc`.

De modo semelhante, pode configurar sistemas centralizados alternativos com outros métodos.

- Integração de utilizador e grupo com o sistema Windows.
  - Aceda a serviços de domínio Windows com os pacotes `winbind` e `libpam_winbind`.
  - Veja winbindd(8) e Integrar Redes MS Windows com Samba.
- Integração de utilizador e grupo com o sistema antigo tipo-Unix.
  - Acesso NIS (originalmente chamado YP) ou NIS+ pelo pacote `nis`.
  - Veja o Linux NIS(YP)/NYS/NIS+ HOWTO.

4.5.3 "Porque o su do GNU não suporta o grupo wheel"

Esta é a famosa frase no fundo da antiga página "`info su`" por Richard M. Stallman. Não se preocupe, o comando `su` atual em Debian utiliza PAM, portanto esse pode recadeiair a habilidade de utilizar `su` ao grupo `root` ao ativar a linha com "`pam_wheel.so`" em "`/etc/pam.d/su`".

4.5.4 Regras de palavra-passe rigorosas

A instalação do pacote `libpam-cracklib` permite-lhe forçar uma regra de palavra-passe mais rigorosa.

Num sistema GNOME típico que instala automaticamente o `libpam-gnome-keyring`, "`/etc/pam.d/common-password`" tem o seguinte aspeto:

```
# here are the per-package modules (the "Primary" block)
password  requisite     pam_cracklib.so retry=3 minlen=8 difok=3
password  [success=1 default=ignore]  pam_unix.so obscure use_authtok try_first_pass  ↩
    yescrypt
# here's the fallback if no module succeeds
password  requisite     pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password  required      pam_permit.so
# and here are more per-package modules (the "Additional" block)
password  optional  pam_gnome_keyring.so
# end of pam-auth-update config
```

4.6 Segurança da autenticação

Nota
A informação aqui **pode não ser suficiente** para as suas necessidades de segurança, mas deverá ser um **bom começo**.

4.6.1 Palavra-passe segura na Internet

Muitos serviços de transporte populares comunicam mensagens incluindo a a autenticação de palavra-passe em texto simples. É má ideia transmitir as palavras-passe em texto simples pela Internet onde podem ser interceptadas. Pode correr estes serviços sobre "Transport Layer Security" (TLS) ou o antecessor dele, "Secure Sockets Layer" (SSL) para assegurar toda a comunicação incluindo a palavra-passe pela encriptação.

nome do serviço inseguro	porto	nome do serviço seguro	porto
`www (http)`	80	`https`	443
`smtp (mail)`	25	`ssmtp (smtps)`	465
`ftp-data`	20	`ftps-data`	989
`ftp`	21	`ftps`	990
`telnet`	23	`telnets`	992
`imap2`	143	`imaps`	993
`pop3`	110	`pop3s`	995
`ldap`	389	`ldaps`	636

Tabela 4.7: Lista de serviços e portos inseguros e seguros

A encriptação custa tempo de CPU. Como uma alternativa amiga para o CPU, pode manter a comunicação em texto simples enquanto segura apenas a palavra-passe com um protocolo de autenticação de segurança como o "Authenticated Post Office Protocol" (APOP) para POP e "Challenge-Response Authentication Mechanism MD5" (CRAM-MD5) para SMTP e IMAP. (Para enviar mensagens de mail pela Internet para o seu servidor de mail a partir do seu cliente de mail, recentemente é popular utilizar o porto 587 para submissão de novas mensagens em vez do tradicional porto 25 do SMTP para evitar o bloqueio do porto 25 pelo provedor de Internet enquanto se autentica com CRAM-MD5.)

A autenticação tradicional baseada em palavra passe ainda sofre de problemas de segurança inerentes tais como a reutilização da palavra passe entre múltiplos sítios web. Veja outros métodos de autenticação para uma solução segura: Palavra passe de uma vez baseada em tempo (TOTP), 2º fator universal (U2F), Autenticação Web (WebAuthn, muito referido como "passkey" ou chave-passe)

4.6.2 Shell Segura

O programa Secure Shell (SSH) disponibiliza comunicações encriptadas e seguras entre duas máquinas sem confiança sobre uma rede insegura com a autenticação de segurança. Consiste no cliente OpenSSH, ssh(1) e no daemon OpenSSH, sshd(8). Este SSH pode ser utilizado como túnel de segurança para protocolos de comunicação inseguros como o POP e X pela Internet com a funcionalidade de reencaminhamento de portos.

O cliente tenta autenticar-se a si próprio a utilizar autenticação baseada na máquina, autenticação de chave pública, autenticação por resposta a desafio, ou autenticação por palavra-passe. O uso de autenticação de chave pública activa o login remoto sem-palavra-passe. Veja Secção 6.3.

4.6.3 Medidas de segurança extra para a Internet

Mesmo quando correr serviços seguros como o Secure Shell (SSH) e servidores de Protocolo de túnel ponto-para-ponto (PPTP), ainda existe hipótese de invasões que utilizam ataques de força bruta à palavra-passe, etc. a partir da Internet. A utilização de politicas de firewall (veja Secção 5.7) juntamente com as seguinte ferramentas de segurança podem melhorar a situação de segurança.

pacote	popcon	tamanho	descrição
`knockd`	V:0.7, I:1.6	110	pequeno daemon port-knock knockd(1) e cliente knock(1)
`fail2ban`	V:101, I:111	2275	banir IPs que causam vários erros de autenticação
`libpam-shield`	V:0.04, I:0.05	115	bloquear atacantes remotos que tentam adivinhar a palavra-passe

Tabela 4.8: Lista de ferramentas para disponibilizar medidas de segurança extra

4.6.4 Tornar a palavra-passe do root segura

Para prevenir que pessoas acedam à sua máquina com privilégios de root, precisa de tomar as seguintes acções.

- Prevenir acesso físico ao dispositivo de armazenamento do sistema (HDD / SSD / ...)
- Bloquear UEFI/BIOS e impedir o arranque a partir de suportes amovíveis
- Definir palavra-passe para sessão interactiva do GRUB
- Bloquear o menu do GRUB de ser editado

Com acesso físico ao dispositivo de armazenamento do sistema, redefinir a palavra-passe é relativamente fácil com os seguintes passos.

1. Mova o dispositivo de armazenamento do sistema para um PC com UEFI/BIOS com arranque a partir de USB.
2. Arranque o sistema com uma média de salvamento (veja Secção 3.2.2).
3. Montar a partição raiz com acesso de leitura/escrita.
4. Editar "`/etc/passwd`" na partição raiz e esvaziar a segunda entrada para a conta `root`.

Se tiver acesso de edição à entrada do menu do GRUB (veja Secção 3.1.2) `grub-rescue-pc` no momento do arranque, é ainda mais fácil com os seguintes passos.

1. Arrancar o sistema com o parâmetro de kernel alterado para algo como "`root=/dev/sda6 rw init=/bin/sh`".
2. Editar "`/etc/passwd`" e tornar a segunda entrada para a conta `root` vazia.
3. Reiniciar o sistema.

A shell de root do sistema está agora acessível sem palavra-passe.

Nota
Após alguém ter acesso à shell de root, pode aceder a tudo no sistema e reiniciar quaisquer palavras-passe no sistema. Mais ainda, pode comprometer a palavra-passe para todas as contas de utilizadores a usar ferramentas de crack de palavra-passe por força bruta como os pacotes `john` e `crack` (veja Secção 9.5.10). Esta palavra-passe descoberta pode levar a comprometer outros sistemas.

A única solução de software razoável para evitar estas preocupações é usar uma partição raiz encriptada por software (ou uma partição "`/etc`" a utilizar dm-crypt e initramfs (veja Secção 9.9). Assim, vai necessitar sempre de palavra-passe para arrancar o sistema.

4.7 Outros controles de acesso

Existem outros controlos de acesso ao sistema para além da autenticação baseada em palavras-passe e das permissões de ficheiros.

Nota
Ver Secção 9.4.16 para restringir a funcionalidade SAK (chave de atenção segura) do kernel.

4.7.1 Listas de controlo de acesso (ACLs)

As ACLs são um superconjunto das permissões regulares, conforme explicado em Secção 1.2.3.

Encontramos ACLs em ação no ambiente de trabalho moderno. Quando um dispositivo de armazenamento USB formatado é montado automaticamente como, por exemplo, "`/media/penguin/USBSTICK`", um utilizador normal, o `penguin`, pode executá-lo:

```
 $ cd /media/penguin
 $ ls -la
total 16
drwxr-x---+ 1 root    root    16 Jan 17 22:55 .
drwxr-xr-x  1 root    root    28 Sep 17 19:03 ..
drwxr-xr-x  1 penguin penguin 18 Jan  6 07:05 USBSTICK
```

"+" na 11ª coluna indica que as ACLs estão em ação. Sem ACLs, um utilizador normal, o `penguin`, não deveria ser capaz de fazer uma lista como esta, uma vez que o `penguin` não está no grupo de `root`. Pode ver as ACLs como:

```
 $ getfacl .
# file: .
# owner: root
# group: root
user::rwx
user:penguin:r-x
group::---
mask::r-x
other::---
```

Aqui:

- "`user::rwx`", "`group::---`", e "`other::---`" correspondem às permissões normais de proprietário, grupo e outros.
- A ACL "`user:penguin:r-x`" permite que um utilizador normal, o `penguin`, tenha permissões "`r-x`". Isso permite que "`ls -la`" liste o conteúdo do diretório.
- A ACL "`mask::r-x`" define o limite superior das permissões.

Consulte "Listas de Controlo de Acesso POSIX em Linux", acl(5), getfacl(1) e setfacl(1) para mais informações.

4.7.2 sudo

O sudo(8) é um programa desenhado para permitir a um administrador de sistema dar privilégios de root limitados a utilizadores e registar a atividade do root. O `sudo` necessita apenas da palavra-passe de um utilizador normal. Instale o pacote `sudo` e active-o ao definir opções em "`/etc/sudoers`". Veja um exemplo de configuração em "`/usr/share/doc/sudo/examples/sudoers`" e Secção 1.1.12.

A minha utilização do `sudo` para o sistema de único utilizador (veja Secção 1.1.12) destina-se a proteger-me da minha própria estupidez. Pessoalmente, Considero utilizar o `sudo` uma melhor alternativa do que utilizar o sistema a partir da conta de root a toda a hora. Por exemplo, o seguinte muda o dono de "*`algum_ficheiro`*" para "*`meu_nome`*".

```
$ sudo chown my_name some_file
```

Claro que se conhecer a palavra-passe de root (como os utilizadores de sistemas Debian auto-instalados conhecem), qualquer comando pode ser executado sob root a partir da conta de qualquer utilizador a usar "`su -c`".

4.7.3 PolicyKit

PolicyKit é um componente do sistema operativo para controlar privilégios a nível global em sistemas operativos tipo-Unix.

Aplicações GUI mais recentes não são desenhadas para correr como processos privilegiados. Estas falam com os processos privilegiados através de PolicyKit para executar operações administrativas.

O PolicyKit limita tais operações a contas de utilizador pertencentes ao grupo `sudo` no sistema Debian.

Veja polkit(8).

4.7.4 Recadeiair acesso a alguns serviços de servidor

Para a segurança do sistema, é uma boa ideia desativar o máximo de programas de serviços possíveis. Isto torna-se crítico em servidores na rede. Ter servidores não utilizados, activados directamente como daemon ou via programa super-server, são considerados riscos de segurança.

Muito programas, como o sshd(8), utilizam controlos de acesso baseados no PAM. Existem muitas maneiras de restringir o acesso a alguns serviços de servidor.

- ficheiros de configuração: "`/etc/default/`*`nome_do_programa`*"
- Configuração Systemd de unidade de serviço para daemon
- PAM (Módulos de autenticação inseríveis)
- "`/etc/inetd.conf`" para super-server
- "`/etc/hosts.deny`" e "`/etc/hosts.allow`" para wrapper de TCP, tcpd(8)
- "`/etc/rpc.conf`" para RPC da Sun
- "`/etc/at.allow`" e "`/etc/at.deny`" para atd(8)
- "`/etc/cron.allow`" e "`/etc/cron.deny`" para crontab(1)
- Firewall de Rede da infraestrutura netfilter

Ver Secção 3.6, Secção 4.5.1, e Secção 5.7.

Dica
os serviços Sun RPC necessitam estar ativos para NFS e outros programas baseados em RPC.

Dica
Se tem problemas com acesso remoto num sistema Debian recente, comente configurações ofensivas tais como "`ALL: PARANOID`" em "`/etc/hosts.deny`" se existirem. (Mas tem que ter cuidado com os riscos de segurança envolvidos com este tipo de acção.)

4.7.5 Caraterísticas de segurança do Linux

O kernel do Linux evoluiu e suporta caraterísticas de segurança que não se encontram nas implementações tradicionais do UNIX.

O Linux suporta atributos alargados que estendem os atributos UNIX tradicionais (ver xattr(7)).

O Linux divide os privilégios tradicionalmente associados ao super utilizador em unidades distintas, conhecidas como capacidades(7), que podem ser ativadas e desativadas independentemente. As capacidades são um atributo por sub-processo desde a versão 2.2 do kernel.

A estrutura do Módulo de Segurança do Linux (LSM) fornece um mecanismo para várias verificações de segurança serem agarradas por novas extensões do kernel. Por exemplo:

- AppArmor
- Linux com segurança reforçada (SELinux)
- Smack (Controlo de Acesso Obrigatório Simplificado Kernel)
- Tomoyo Linux

Uma vez que estas extensões podem apertar mais o modelo de privilégios do que as políticas normais do modelo de segurança tipo Unix, até o poder de root pode ser restringido. Aconselha-se a ler o documento da estrutura do Módulo de Segurança Linux (LSM) em kernel.org.

Os namespaces Linux envolvem um recurso global do sistema numa abstração que faz parecer aos processos dentro do namespace que têm a sua própria instância isolada do recurso global. As alterações ao recurso global são visíveis para outros processos que são membros do espaço de nomes, mas são invisíveis para outros processos. Desde a versão 5.6 do kernel, existem 8 tipos de espaços de nomes (veja namespaces(7), unshare(1), nsenter(1)).

A partir da Debian 11 Bullseye (2021), a Debian usa uma hierarquia de cgroups unificada (a.k.a. cgroups-v2).

Exemplos de utilização de namespaces com cgroups para isolar os seus processos e permitir o controlo de recursos:

- Systemd. Ver Secção 3.3.1.
- Sandbox environment. Ver Secção 7.7.
- Contentores Linux como o Docker, LXC. Ver Secção 9.11.

Estas funcionalidades não podem ser realizadas por Secção 4.1. Estes tópicos avançados estão, na sua maioria, fora do âmbito deste documento introdutório.

Capítulo 5

Configuração de rede

Dica
Para um guia moderno específico Debian sobre rede, veja O Livro de Mão do Administrador Debian — Configurar a Rede.

Dica
Sob systemd, pode ser usado o networkd para gerir as redes. Veja systemd-networkd(8).

5.1 A infra-estrutura de rede básica

Vamos rever a infra-estrutura básica de rede do sistema Debian moderno.

5.1.1 A resolução de nome de máquina

A resolução de nome de máquina também é suportada atualmente pelo mecanismo NSS (Name Service Switch). O fluxo desta resolução é o seguinte.

1. O ficheiro "`/etc/nsswitch.conf`" com "`hosts: files dns`" dicta a ordem de resolução do nome de máquina. (Isto substitui a funcionalidade antiga de "`order`" em "`/etc/host.conf`".)
2. O método `files` é invocado primeiro. Se o nome de máquina for encontrado no ficheiro "`/etc/hosts`", devolve um endereço válido e termina. (O ficheiro "`/etc/host.conf`" contém "`multi on`".)
3. O método `dns` é invocado. Se o nome de máquina é encontrado pela consulta ao Internet Domain Name System (DNS) identificado pelo ficheiro "`/etc/resolv.conf`", devolve-le um endereço válido e termina.

Uma estação de trabalho típica pode ser instalada com seu nome de host definido como, por exemplo, "*host_name*" e seu nome de domínio opcional definido como uma string vazia. Então, o "`/etc/hosts`" tem o seguinte aspeto.

```
127.0.0.1 localhost
127.0.1.1 host_name

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

pacotes	popcon	tamanho	tipo	descrição
network-manager	V:427, I:474	7365	config::NM	NetworkManager (daemon): gere a rede automaticamente
network-manager-gnome	V:36, I:147	19	config::NM	NetworkManager (frontend do GNOME)
netplan.io	V:2.1, I:8.4	340	config::NM+networkd	Netplan (gerador): Interface unificada e declarativa para os backends NetworkManager e systemd-networkd
ifupdown	V:649, I:973	204	config::ifupdown	ferramenta standard para ativar e desativar a rede (especifico de Debian)
pppoeconf	V:0.2, I:3.9	176	config::helper	ajudante de configuração para ligação PPPoE
wpasupplicant	V:421, I:521	3896	config::helper	suporte de cliente para WPA e WPA2 (IEEE 802.11i)
wpagui	V:0.2, I:1.2	779	config::helper	Cliente GUI Qt para o wpa_supplicant
wireless-tools	V:193, I:262	232	config::helper	ferramentas para manipular Extensões Wireless do Linux
iw	V:40, I:478	332	config::helper	ferramenta para configuração de dispositivos sem fios Linux
iproute2	V:781, I:987	4268	config::iproute2	iproute2, IPv6 e outras configurações avançadas de rede: ip(8), tc(8), etc
iptables	V:381, I:635	2408	config::Netfilter	ferramentas administrativas para filtragem de pacotes e NAT (Netfilter)
nftables	V:266, I:880	189	config::Netfilter	ferramentas de administração para filtragem de pacotes e NAT (Netfilter) (sucessor de {ip,ip6,arp,eb}tables)
iputils-ping	V:206, I:998	188	teste	teste de acessibilidade de rede de uma máquina remota pelo nome-de-máquina ou endereço IP (iproute2)
iputils-arping	V:2, I:17	53	teste	teste de acessibilidade de rede de uma máquina remota especificado pelo endereço ARP
iputils-tracepath	V:2, I:19	50	teste	rastreia o caminho de rede até uma máquina remota
ethtool	V:98, I:251	1093	teste	mostra ou altera as definições de um dispositivo Ethernet
mtr-tiny	V:4, I:38	182	test::low-level	rastreia o caminho de rede até uma máquina remota (curses)
mtr	V:4, I:40	231	test::low-level	rastreia o caminho de rede até uma máquina remota (curses e GTK)
gnome-nettool	V:0.5, I:8.4	2480	test::low-level	ferramentas para operações comuns de informação de rede (GNOME)
nmap	V:25, I:182	4934	test::low-level	mapeamento de rede / sondagem de portos (Nmap, consola)
tcpdump	V:20, I:177	1346	test::low-level	analisador de tráfego de rede (Tcpdump, consola)
wireshark	V:4, I:39	17068	test::low-level	analisador de tráfego de rede (Wireshark, GTK)
tshark	V:2, I:23	433	test::low-level	analisador de tráfego de rede (consola)
tcptrace	V:0.2, I:1.5	407	test::low-level	produz um sumário das ligações a partir da saída do `tcpdump`
dnsutils	V:5, I:146	23	test::low-level	clientes de rede disponibilizados com BIND: nslookup(8), nsupdate(8), dig(8)
dlint	V:0.1, I:2.0	52	test::low-level	verifica informação de zona DNS a usar pesquisas do servidor de nomes
dnstracer	V:0.1, I:1.1	63	test::low-level	rastreia uma cadeia de servidores DNS até à fonte

Tabela 5.1: Lista de ferramentas de configuração de rede

Cada linha começa por um endereço IP e é seguida pelo hostname associado.

O endereço IP `127.0.1.1` na segunda linha deste exemplo pode não ser encontrada nalguns outros sistemas tipo-Unix. O Instalador Debian cria esta entrada para um sistema sem um endereço IP permanente como contenção para algum software (p.e., GNOME) conforme é documentado no bug #719621.

O *host_name* corresponde ao nome de anfitrião definido no ficheiro "`/etc/hostname`" (ver Secção 3.8.1).

Para um sistema com um endereço IP permanente, esse endereço IP permanente deve ser utilizado aqui em vez do `127.0.1.1`.

Para um sistema com um endereço IP permanente e um nome de domínio totalmente qualificado (FQDN) disponibilizado pelo Sistema de Nomes de Domínio (DNS), o *nome_máquina* e *nome_domínio* canónicos devem ser utilizados em vez de apenas *nome_máquina*.

O "`/etc/resolv.conf`" é um ficheiro estático se o pacote `resolvconf` não estiver instalado. Se instalado, é uma ligação simbólica. De qualquer modo, contém informação que inicializa as rotinas de resolução de nomes. Se o DNS existir no IP="`192.168.11.1`", contém o seguinte.

```
nameserver 192.168.11.1
```

O pacote `resolvconf` faz deste "`/etc/resolv.conf`" uma ligação simbólica e gere o conteúdo dele automaticamente pelos scripts hook.

Para a estação de trabalho PC no ambiente típico de LAN adhoc, o nome do anfitrião pode ser resolvido através do Multicast DNS (mDNS), para além dos métodos básicos de `files` e `dns`.

- Avahi disponibiliza uma infraestrutura para Multicast DNS Service Discovery em Debian.
- É o equivalente de Apple Bonjour / Apple Rendezvous.
- O pacote de plugin `libnss-mdns` disponibiliza resolução de nomes de máquinas via mDNS para a funcionalidade Name Service Switch (NSS) do GNU da Biblioteca C do GNU (glibc).
- O ficheiro "`/etc/nsswitch.conf`" deve ter uma estrofe como "`hosts: files mdns4_minimal [NOTFOUND=retur dns`" (veja `/usr/share/doc/libnss-mdns/README.Debian` para outras configurações).
- Um nome de máquina sufixado com ".local" de domínio pseudo-nível de topo é resolvido enviando uma mensagem de consulta mDNS num pacote UDP multicast utilizando o endereço IPv4 "`224.0.0.251`" ou o endereço IPv6 "`FF02::FB`".

Nota
A expansão Top-Level Domains genéricos (gTLD) no Domain Name System está em desenvolvimento. Tenha cuidado com a colisão de nomes quando escolher um nome de domínio usado apenas dentro da LAN.

Nota
O uso de pacotes como o `libnss-resolve` em conjunto com o `systemd-resolved`, ou `libnss-myhostname`, ou `libnss-mymachine`, com listagens correspondentes na linha "`hosts`" no ficheiro "`/etc/nsswitch.conf`" pode sobrepor-se à configuração de rede tradicional discutida em cima. Veja nss-resolve(8), systemd-resolved(8), nss-myhostname(8), e nss-mymachines(8) para mais informação.

5.1.2 O nome da interface de rede

O systemd usa "Predictable Network Interface Names" como "`enp0s25`".

5.1.3 A gama de endereços de rede para a LAN

Vamos nos lembrar das gamas de endereços do IPv4 de 32 bits em cada classe reservada para utilização em redes de área local (LANs) pelo rfc1918. É garantido que estes endereços não estejam em conflito com quaisquer endereços na Internet.

Nota
Os endereços IP escritos com dois pontos são endereços IPv6, por exemplo, "`::1`" para `máquina local`.

Classe	endereços de rede	máscara de rede	máscara de rede /bits	número de sub-redes
A	10.x.x.x	255.0.0.0	/8	1
B	172.16.x.x — 172.31.x.x	255.255.0.0	/16	16
C	192.168.0.x — 192.168.255.x	255.255.255.0	/24	256

Tabela 5.2: Lista de gamas de endereços de rede

Nota
Se for atribuído um destes endereços a uma máquina, então essa máquina não poderá aceder à Internet directamente mas terá de aceder através de uma gateway que age como um proxy para serviços individuais ou então fará Network Address Translation(NAT). O router de banda larga geralmente executa NAT para o ambiente LAN do consumidor.

5.1.4 O suporte a dispositivos de rede

Apesar da maioria dos dispositivos de hardware serem suportados pelo sistema Debian, existem alguns dispositivos de rede que necessitam de firmware não-livre DFSG para os suportar. Por favor veja Secção 9.10.5.

5.2 A configuração moderna de rede para desktop

As interfaces de rede são tipicamente iniciadas em "`networking.service`" para a interface `lo` e "`NetworkManager.ser`
para as outras interfaces em sistemas Debian modernos sob `systemd`.

Debian pode gerir a ligação de rede através de software daemon de gestão como o NetworkManager (NM) (gestor de rede e pacotes associados).

- Vêm com as suas próprias GUIs e programas de linha de comandos como as suas interfaces de utilizador.
- Vêm com os daemons próprios como os backends deles.
- Permitem uma ligação fácil do seu sistema à Internet.
- Permitem gestão fácil de configuração de redes com e sem fios.
- Permitem-nos configurar a rede independentemente do pacote legacy `ifupdown`.

Nota
Não utilize estas ferramentas de configuração de rede automáticas em servidores. Estas são destinadas principalmente para os utilizadores móveis em portáteis.

Estas ferramentas modernas de configuração de rede necessitam ser correctamente configuradas para evitar entrarem em conflito com o pacote legacy `ifupdown` e o ficheiro de configuração dele "`/etc/network/interfaces`".

5.2.1 Ferramentas GUI de configuração de rede

As documentações oficiais para o NM em Debian são disponibilizadas em "`/usr/share/doc/network-manager/READM`

Essencialmente, a configuração de rede para desktop é feita como a seguir.

1. Tornar o utilizador de ambiente de trabalho, p.e. `foo`, pertencente ao grupo "`netdev`" com o seguinte (Em alternativa, em ambientes de trabalho modernos como o GNOME e o KDE, faça-o automaticamente através de D-bus).

```
$ sudo usermod -a -G netdev foo
```

2. Mantenha a configuração de "`/etc/network/interfaces`" tão simples como o seguinte.

```
auto lo
iface lo inet loopback
```

3. Reinicie o NM da seguinte forma.

```
$ sudo systemctl restart NetworkManager
```

4. Configure a sua rede através da GUI.

Nota
Apenas as interfaces que **não** estão listadas em "`/etc/network/interfaces`" são geridas pelo NM para evitar conflitos com o `ifupdown`.

Dica
Se desejar estender as capacidades de configuração de rede do NM, por favor procure módulos plug-in apropriados e pacotes suplementares como os `network-manager-openconnect`, `network-manager-openvpn-gnome`, `network-manager-pptp-gnome`, `mobile-broadband-provider-info`, `gnome-bluetooth`, etc.

5.3 A moderna configuração de rede sem GUI

Sob systemd, a rede pode ser configurada em `/etc/systemd/network/`. Veja systemd-resolved(8), resolved.conf(5) e systemd-networkd(8).

Isto permite a configuração moderna de rede sem GUI.

Uma configuração de cliente DHCP pode ser definida ao criar `/etc/systemd/network/dhcp.network`". Ex.:

```
[Match]
Name=en*

[Network]
DHCP=yes
```

Uma configuração de rede estática pode ser definida ao criar "`/etc/systemd/network/static.network`". Ex.:

```
[Match]
Name=en*

[Network]
Address=192.168.0.15/24
Gateway=192.168.0.1
```

5.4 A configuração moderna de rede para nuvem

A configuração de rede moderna para a nuvem pode usar os pacotes `cloud-init` e `netplan.io` (ver Secção 3.8.4).

O pacote `netplan.io` suporta `systemd-networkd` e `NetworkManager` como seus backends de configuração de rede, e permite a configuração de rede usando dados YAML. Quando você altera o YAML:

- Execute o comando "`netplan generate`" para gerar toda a configuração de backend necessária a partir de YAML.
- Execute o comando "`netplan apply`" para aplicar a configuração gerada aos backends.

Veja "Documentação Netplan", netplan(5), netplan-generate(8) e netplan-apply(8).

Consulte também "Documentação de Inicialização na Nuvem" (especialmente em "Fontes de configuração" e "Netplan Passthrough") para saber como o `cloud-init` pode integrar a configuração do `netplan.io` com fontes de dados alternativas.

5.4.1 A configuração moderna de rede para nuvem com DHCP

A configuração de um cliente DHCP pode ser definida através da criação de um ficheiro de dados "`/etc/netplan/50-dhcp`

```
network:
  version: 2
  ethernets:
    all-en:
      match:
        name: "en*"
      dhcp4: true
      dhcp6: true
```

5.4.2 A configuração moderna de rede para nuvem com IP estático

Uma configuração de rede estática pode ser definida através da criação de um ficheiro de dados"`/etc/netplan/50-stati`

```
network:
  version: 2
  ethernets:
    eth0:
      addresses:
        - 192.168.0.15/24
      routes:
        - to: default
          via: 192.168.0.1
```

5.4.3 A configuração moderna de rede para nuvem com Network Manager

A configuração do cliente de rede utilizando a infraestrutura do Network Manager pode ser definida através da criação de um ficheiro de dados "`/etc/netplan/00-network-manager.yaml`":

```
network:
  version: 2
  renderer: NetworkManager
```

5.5 A configuração de rede de baixo nível

Para a configuração de rede de baixo nível no Linux, use os programas iproute2 (ip(8), ...) .

5.5.1 Comandos iproute2

Os comandos Iproute2 oferecem capacidades completas de configuração de rede de baixo nível. Aqui está uma tabela de traduções dos comandos net-tools obsoletos para os novos comandos iproute2 etc.

ferramentas de rede obsoletas	**novo iproute2 etc.**	**manipulação**
ifconfig(8)	`ip addr`	endereço de protoco (IP ou IPv6) num dispositivo
route(8)	`ip route`	entrada na tabela de rotas
arp(8)	`ip neigh`	entrada na cache ARP ou NDISC
`ipmaddr`	`ip maddr`	endereço multicast
`iptunnel`	`ip tunnel`	túnel sobre IP
nameif(8)	ifrename(8)	nomeia as interfaces de rede baseadas no endereço MAC
mii-tool(8)	ethtool(8)	Definições de dispositivo Ethernet

Tabela 5.3: Tabela de tradução dos comandos obsoletos `net-tools` para os novos comandos `iproute2`

Ver ip(8) e Encaminhamento avançado Linux & Controlo de Tráfego.

5.5.2 Operações de rede seguras de baixo nível

Pode usar comandos de rede de baixo nível como a seguir em segurança pois eles não mudam a configuração de rede.

comando	**descrição**
`ip addr show`	mostra o estado de ligação e endereço das interfaces ativas
`route -n`	mostra toda a tabela de rotas em endereços numéricos
`ip route show`	mostra toda a tabela de rotas em endereços numéricos
`arp`	mostra o conteúdo atual das tabelas de cache ARP
`ip neigh`	mostra o conteúdo atual das tabelas de cache ARP
`plog`	mostra o log do daemon ppp
`ping yahoo.com`	verifica a ligação de Internet para "`yahoo.com`"
`whois yahoo.com`	verifica quem registou "`yahoo.com`" na base de dados de domínios
`traceroute yahoo.com`	rastreia a ligação Internet até "`yahoo.com`"
`tracepath yahoo.com`	rastreia a ligação Internet até "`yahoo.com`"
`mtr yahoo.com`	rastreia a ligação Internet até "`yahoo.com`" (repetidamente)
`dig [@dns-server.com] example.com [{a|mx|any}]`	verifica os registos DNS de "`exemplo.com`" pelo "`servidor-dns.com`" para um registo "a", "mx", ou "any"
`iptables -L -n`	verifica o filtro de pacotes
`ss -a`	procura todos os portos abertos
`ss -l`	procura portos a escutar
`ss -l -t -n`	procura portos TCP a escutar (numérico)
`dlint example.com`	verifica a informação da zona DNS de "`exemplo.com`"

Tabela 5.4: Lista de comandos de rede de baixo nível

Dica
Algumas destas ferramentas de configuração de baixo nível residem em "`/usr/sbin/`". Pode necessitar de escrever o caminho do comando completo tal como "`/usr/sbin/ifconfig`" ou adicionar "`/usr/sbin`" à lista "`$PATH`" no seu "`~/.bashrc`".

5.6 Optimização da rede

A optimização de rede genérica está para além do objetivo desta documentação. Apenas toco em assuntos pertinentes às ligações de grau de consumidor.

pacotes	popcon	tamanho	descrição
iftop	V:6, I:85	93	mostra informação da utilização de largura de banda numa interface de rede
iperf	V:2, I:32	431	ferramenta de medição da largura de banda do Protocolo Internet
ifstat	V:0.5, I:5.3	52	Monitorização estatística Interface
bmon	V:2, I:19	154	monitor de largura de banda portável e estimador de taxas
ethstatus	V:0.2, I:2.3	41	script que mede rapidamente a transferência efectiva de um dispositivo de rede
bing	V:0.08, I:0.47	80	testador de largura de banda empírica estocástica
bwm-ng	V:1.0, I:9.1	95	monitor de largura de banda pequeno e simples baseado em consola
ethstats	V:0.05, I:0.36	21	monitor de estatísticas de Ethernet baseado em consola
ipfm	V:0.04, I:0.09	78	ferramenta de análise de largura de banda

Tabela 5.5: Lista de ferramentas de optimização de rede

5.6.1 Encontrar o MTU óptimo

Normalmente, o NM define automaticamente a Unidade Máxima de Transmissão (MTU) ideal.

Nalgumas ocasiões, pode desejar definir o MTU manualmente após experiências com o ping(8) com a opção "`-M do`" para enviar um pacote ICMP com vários tamanhos de pacotes de dados. MTU é o tamanho máximo do pacote de dados sucessivo sem fragmentação de IP mais 28 bytes para o IPv4 e mais 48 bytes para o IPv6. Por exemplo, o seguinte indica que a MTU da ligação IPv4 é 1460 e a MTU da ligação IPv6 é 1500.

```
$ ping -4 -c 1 -s $((1500-28)) -M do www.debian.org
PING  (149.20.4.15) 1472(1500) bytes of data.
ping: local error: message too long, mtu=1460

---  ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

$ ping -4 -c 1 -s $((1460-28)) -M do www.debian.org
PING  (130.89.148.77) 1432(1460) bytes of data.
1440 bytes from klecker-misc.debian.org (130.89.148.77): icmp_seq=1 ttl=50 time=325 ms

---  ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 325.318/325.318/325.318/0.000 ms
```

```
$ ping -6 -c 1 -s $((1500-48)) -M do www.debian.org
PING www.debian.org(mirror-csail.debian.org (2603:400a:ffff:bb8::801f:3e)) 1452 data bytes
1460 bytes from mirror-csail.debian.org (2603:400a:ffff:bb8::801f:3e): icmp_seq=1 ttl=47 ↩
    time=191 ms

--- www.debian.org ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 191.332/191.332/191.332/0.000 ms
```

Este processo é a descoberta do (PMTU) Caminho MTU (RFC1191) e o comando tracepath(8) pode automatizar isto.

ambiente de rede	MTU	racional
Ligação Dial-up (IP: PPP)	576	standard
Ligação Ethernet (IP: DHCP ou fixo)	1500	standard e predefinido

Tabela 5.6: Regras básicas para o valor MTU óptimo

Adicionalmente a estas regras básicas, deve saber o seguinte.

- Qualquer utilização de métodos de túnel (VPN etc.) pode reduzir o MTU óptimo pelo excesso que adicionam.
- O valor MTU não deve exceder o valor PMTU determinado experimentalmente.
- O maior valor MTU é geralmente melhor quando são conhecidas outras limitações.

O tamanho de segmento máximo (MSS) é usado como uma alternativa ao tamanho do pacote. As relações entre MSS e MTU são as seguintes.

- MSS = MTU - 40 para IPv4
- MSS = MTU - 60 para IPv6

Nota
A optimização baseada no iptables(8) (veja Secção 5.7) pode apertar o tamanho do pacote pelo MSS e é útil para o router. Veja ”TCPMSS” em iptables(8).

5.6.2 Optimização WAN TCP

A taxa de transferência TCP pode ser maximizada ajustando os parâmetros de tamanho do buffer TCP como em ”TCP tuning” para a moderna WAN de alta largura de banda e alta latência. Até agora, as configurações padrão atuais do Debian servem bem até mesmo para minha LAN conectada pelo rápido serviço FTTP de 1G bps.

5.7 Infraestrutura netfilter

Netfilter disponibiliza uma infra-estrutura para firewall de estado e tradução de endereços de rede (NAT) com módulos do kernel Linux (veja Secção 3.10).

Veja O projeto netfilter.org ”nftables” para o comando nft(8) de espaço de utilizador mais recente no pacote `nftables`. (Kernel linux >= 3.13)

pacotes	popcon	tamanho	descrição
nftables	V:266, I:880	189	ferramentas de administração para filtragem de pacotes e NAT (Netfilter) (sucessor de {ip,ip6,arp,eb}tables)
iptables	V:381, I:635	2408	ferramentas de administração para netfilter (iptables(8) para IPv4, ip6tables(8) para IPv6)
arptables	V:0.1, I:1.6	102	ferramentas de administração para netfilter (arptables(8) para ARP)
ebtables	V:14, I:23	276	ferramentas de administração para netfilter (ebtables(8) para criação de pontes Ethernet)
iptstate	V:0.1, I:1.6	121	monitoriza continuamente o estado do netfilter (semelhante ao top(1))
ufw	V:83, I:109	862	Uncomplicated Firewall (UFW) é um programa para gerir filtros de pacotes de rede
gufw	V:6, I:11	3682	interface gráfica do utilizador para a Uncomplicated Firewall (UFW)
firewalld	V:15, I:22	2581	O firewalld é um programa de filtro de pacotes gerido dinamicamente com suporte para zonas de rede
firewall-config	V:0.8, I:3.2	1096	interface gráfica do utilizador para o firewalld
shorewall-init	V:0.17, I:0.33	88	Inicialização de Shoreline Firewall
shorewall	V:2.1, I:4.7	3090	Shoreline Firewall, gerador de ficheiro de configuração netfilter
shorewall-lite	V:0.03, I:0.06	71	Shoreline Firewall, gerador de ficheiro de configuração (versão leve) netfilter
shorewall6	V:0.6, I:1.2	1334	Shoreline Firewall, gerador de ficheiro de configuração (versão IPv6) netfilter
shorewall6-lite	V:0.01, I:0.02	71	Shoreline Firewall, gerador de ficheiro de configuração (IPv6, versão light) netfilter

Tabela 5.7: Lista de ferramentas de firewall

Veja O projeto netfilter.org ”iptables” para o iptables(8) de espaço de utilizador antigo e comandos semelhantes no `iptables` e pacotes semelhantes. (Kernel linux >= 2.4.x)

Pode configurar manualmente e interativamente o netfilter a partir da shell, gravar o estado dele com ”`nft list ruleset > ficheiro`” ou iptables-save(8) e restaurá-lo via script init com ”`nft -f ficheiro`” ou iptables-restore(8) após o reiniciar do sistema.

Scripts de ajuda de configuração como o shorewall facilitam este processo.

Ver documentação em Documentação Netfilter (ou em ”`/usr/share/doc/iptables/html/`”).

- Linux Networking-concepts HOWTO
- Linux 2.4 Packet Filtering HOWTO
- Linux 2.4 NAT HOWTO
- nftables HOWTO

Capítulo 6

Aplicações de rede

Após estabelecer a ligação de rede (veja Capítulo 5), pode executar varias aplicações de rede.

Dica
Para um guia moderno específico Debian para a infraestrutura de rede, leia O Livro de Mão do Administrador Debian — Infrastructure de Rede.

Dica
Se ativar a ”Verificação de 2 Passos” com alguns ISP, precisa de obter uma palavra-passe de aplicação para aceder aos serviços POP e SMTP do seu programa. Pode precisar de aprovar o seu IP de máquina com antecedência.

6.1 Navegadores web

Existem muitos pacotes de navegadores web para aceder a conteúdos remotos com Hypertext Transfer Protocol (HTTP).

pacote	popcon	tamanho	tipo	descrição do explorador web
chromium	V:31, I:100	320285	X	Chromium, (browser open-source da Google)
firefox	V:15, I:21	308351	, ,	Firefox, (navegador de código aberto da Mozilla, apenas disponível no Debian Unstable)
firefox-esr	V:186, I:425	299827	, ,	Firefox ESR, (Lançamento de Suporte Estendido do Firefox)
epiphany-browser	V:2, I:10	6735	, ,	GNOME, Altamente conforme, Epiphany
konqueror	V:26, I:111	8179	, ,	KDE, Konqueror
dillo	V:0.4, I:3.9	2203	, ,	Dillo, (navegador leve, baseado em FLTK)
w3m	V:10, I:129	2853	texto	w3m
lynx	V:29, I:455	2031	, ,	Lynx
elinks	V:2, I:15	1791	, ,	ELinks
links	V:2, I:20	2321	, ,	Ligações (apenas texto)
links2	V:0.8, I:9.4	5466	gráficos	Ligações (gráficos de consola sem X)

Tabela 6.1: Lista de exploradores web

6.1.1 Falsificação da cadeia User-Agent

Para aceder a alguns sítios Web demasiado restritivos, poderá ser necessário falsificar a cadeia User-Agent devolvida pelo programa de navegação Web. Ver:

- MDN Web Docs: userAgent
- Chrome Developers: Substituir a cadeia de caracteres do agente do utilizador
- Como alterar o seu agente do utilizador
- Como alterar o User-Agent no Chrome, Firefox, Safari e outros
- Como alterar o agente do utilizador do seu navegador sem instalar quaisquer extensões
- Como alterar o agente do utilizador no Gnome Web (epiphany)

6.1.2 Extensão do navegador

Todos os navegadores GUI modernos suportam extensões de navegador baseadas em código-fonte e estão a tornar-se normalizadas como extensões Web.

6.2 O sistema de correio electrónico (mail)

Esta secção centra-se em estações de trabalho móveis típicas com ligações à Internet de nível de consumidor.



Cuidado
Se estiver a configurar um servidor de mail para trocar mail directamente com a Internet, deverá fazer melhor do que ler esta documentação elementar.

6.2.1 Noções básicas de mail

Uma mensagem de email consiste em três componentes, o envelope da mensagem, o cabeçalho da mensagem e o corpo da mensagem.

- A informação ”To” e ”From” no envelope da mensagem é utilizada pelo SMTP para entregar o email. (A informação de ”From” no envelope da mensagem também é chamada de endereço bounce, From_, etc.).
- A informação ”To” e ”From” no cabeçalho da mensagem é mostrada pelo cliente de email. (Embora seja vulgar que sejam os mesmo do envelope da mensagem nem sempre é o caso.)
- O formato da mensagem de correio eletrónico, que abrange os dados do cabeçalho e do corpo, é alargado pelas Extensões de Correio Internet para Fins Múltiplos (MIME) do texto ASCII simples para outras codificações de caracteres, bem como para anexos de áudio, vídeo, imagens e programas de aplicação.

Os clientes de correio eletrónico completos baseados em GUI oferecem todas as funções seguintes utilizando a configuração intuitiva baseada em GUI.

- Cria e interpreta os dados do cabeçalho e do corpo da mensagem utilizando as Extensões de Correio Internet para Fins Múltiplos (MIME) para lidar com o tipo de dados e a codificação do conteúdo.

- Ele se autentica nos servidores SMTP e IMAP do ISP usando a autenticação de acesso básico antiga ou o moderno OAuth 2.0. (Para o OAuth 2.0, defina-o através das definições do ambiente de trabalho. Por exemplo, ”Definições” -> ”Contas Online”.)
- Envia a mensagem para o servidor SMTP smarthost do ISP que está a ouvir a porta de envio de mensagens (587).
- Recebe a mensagem armazenada no servidor do ISP a partir da porta TLS/IMAP4 (993).
- Pode filtrar mensagens de correio eletrónico pelos seus atributos.
- Pode oferecer funcionalidades adicionais: Contactos, Calendário, Tarefas, Memorandos.

pacote	popcon	tamanho	tipo
`evolution`	V:27, I:229	489	Programa de GUI X (GNOME, conjunto groupware)
`thunderbird`	V:39, I:102	303329	Programa X GUI (GTK, Mozilla Thunderbird)
`kmail`	V:43, I:103	25265	Programa de GUI X (KDE)
`mutt`	V:10, I:82	7334	programa de terminal de caracteres provavelmente usado com o `vim`
`mew`	V:0.01, I:0.14	2319	programa de terminal de caracteres sob `(x)emacs`

Tabela 6.2: Lista de agentes utilizador de mail (MUA)

6.2.2 Limitação do serviço de correio moderno

Os serviços de correio modernos estão sujeitos a algumas limitações para minimizar a exposição aos problemas de spam (correio eletrónico não desejado e não solicitado).

- Não é realista correr um servidor SMTP numa rede de ligação doméstica para enviar mail diretamente para a máquina remota de forma confiável.
- Um correio pode ser rejeitado por qualquer anfitrião a caminho do destino, a menos que pareça tão autêntico quanto possível.
- Não é realista para a confiança da máquina remota, esperar que um único smarthost envie mails com um endereço de mail fonte não relacionado.

Isto deve-se ao facto de:

- As ligações da porta SMTP (25) dos anfitriões servidos pela rede de nível de consumo à Internet estão bloqueadas.
- As ligações da porta SMTP (25) estão bloqueadas a partir da Internet aos anfitriões servidos pela rede ao nível do consumidor.
- As mensagens de saída dos anfitriões servidos pela rede ao nível do consumidor para a Internet só podem ser enviadas através da porta de envio de mensagens (587).
- Técnicas Anti-spam tais como DomainKeys Identified Mail (DKIM), Sender_Policy_Framework (SPF) e Domain-based Message Authentication, Reporting and Conformance (DMARC) são muito usadas para a filtragem de email.
- O serviço DomainKeys Identified Mail pode ser disponibilizado para o email que envia através do smarthost.
- O sistema anfitrião inteligente pode reescrever o endereço de correio de origem no cabeçalho da mensagem para a sua conta de correio, para evitar a falsificação de endereços de correio eletrónico.

6.2.3 Expectativa histórica do serviço de correio

Alguns programas em Debian esperam aceder ao comando `/usr/sbin/sendmail` para enviar emails como predefinição ou definição personalizada uma vez que o serviço de mail num sistema UNIX funcionava historicamente como:

- Uma mensagem de correio eletrónico é criada como um ficheiro de texto.
- O correio eletrónico é enviado para o comando `/usr/sbin/sendmail`.
- Para o endereço de destino no mesmo host, o comando `/usr/sbin/sendmail` faz a entrega local do e-mail, anexando-o ao arquivo `/var/mail/$username`.
  - Comandos que esperam esta funcionalidade: `apt-listchanges`, `cron`, `at`, ...
- Para o endereço de destino no anfitrião remoto, o comando `/usr/sbin/sendmail` efetua a transferência remota do correio eletrónico para o anfitrião de destino encontrado pelo registo MX do DNS utilizando SMTP.
  - Comandos que esperam esta funcionalidade: `popcon`, `reportbug`, `bts`, ...

6.2.4 Agente de transporte de mail (MTA)

As estações de trabalho móveis Debian podem ser configuradas apenas com clientes de email baseados em GUI completos sem o programa de agente de transferência de email (MTA) após o Debian 12 Bookworm.

Debian tradicionalmente instalou algum programa MTA para suportar programas que esperam o comando `/usr/sbin/send`. Tal MTA em estações de trabalho móveis tem de lidar com Secção 6.2.2 e Secção 6.2.3.

Para estações de trabalho móveis, a escolha típica de MTA é o `exim4-daemon-light` ou o `postfix` com a opção de instalação "Mail sent by smarthost; received via SMTP or fetchmail" selecionada. Estes são MTAs leves que respeitam "`/etc/aliases`".

Dica
Configurar o `exim4` para enviar o correio da Internet através de múltiplos smarthosts correspondentes para múltiplos endereços de e-mail de origem não é trivial. Se você precisa de tal capacidade para alguns programas, configure-os para usar `msmtp`, que é fácil de configurar para múltiplos endereços de e-mail de origem. Em seguida, deixe o MTA principal apenas para um único endereço de email.

6.2.4.1 A configuração do exim4

Para mail de Internet através de smarthost, (re)configura o pacote `exim4-*` conforme o seguinte:

```
$ sudo systemctl stop exim4
$ sudo dpkg-reconfigure exim4-config
```

Escolha "mail enviado por smarthost; recebido via SMTP ou fetchmail" para "Configuração geral do tipo de mail".

Defina "nome de mail do sistema:" para a predefinição dele como o FQDN (veja Secção 5.1.1).

Defina "Endereço IP onde escutar ligações SMTP recebidas:" à predefinição dele como "127.0.0.1 ; ::1".

Desconfigure o conteúdo de "Outros destinos para o qual o mail é aceite:".

Desconfigure o conteúdo de "Máquinas para retransmitir mail para:".

Defina "Endereço IP ou nome de máquina do smarthost de envio:" para "smtp.nome-de-máquina.domínio:587".

Escolha "Não" para "Esconder o nome de mail local para o mail enviado?". (Em vez disso, use "`/etc/email-addresses`" como em Secção 6.2.4.3.)

Responda a "Mínimizar a quantidade de consultas DNS (Chamar-a-Pedido)?" como uma das seguintes.

pacote	popcon	tamanho	descrição
exim4-daemon-light	V:200, I:209	1776	Agente de transporte de mail Exim4 (MTA: predefinido em Debian)
exim4-daemon-heavy	V:4.8, I:4.9	1954	Agente de transporte de correio Exim4 (MTA: alternativa flexível)
exim4-base	V:205, I:214	1685	Documentação do Exim4 (texto) e ficheiros comuns
exim4-doc-html	I:0.98	3809	Documentação do Exim4 (html)
exim4-doc-info	I:0.55	649	Documentação do Exim4 (info)
postfix	V:102, I:107	4274	Agente de transporte de mail Postfix (MTA: alternativa segura)
postfix-doc	I:4.3	5028	Documentação do Postfix (html+texto)
sasl2-bin	V:4.5, I:9.9	381	Implementação Cyrus SASL API (postfix suplementar para SMTP AUTH)
cyrus-sasl2-doc	I:0.68	2140	Cyrus SASL - documentação
msmtp	V:9, I:15	822	MTA leve
msmtp-mta	V:7.2, I:9.1	139	MTA leve (extensão de compatibilidade do sendmail para `msmtp`)
nullmailer	V:7.4, I:7.7	483	Elimina o MTA, sem correio local
ssmtp	V:3.9, I:5.9	134	Elimina o MTA, sem correio local
sendmail-bin	V:10, I:10	1954	MTA completo (apenas se já estiver familiarizado)
git-email	V:0.3, I:9.3	1229	git-send-email(1) programa para enviar uma série de e-mails de patches

Tabela 6.3: Lista de pacotes básicos relacionados com o agente de transporte de correio

- ”Não” se o sistema estiver ligado à Internet enquanto arranca.
- ”Sim” se o sistema **não** está ligado à Internet enquanto arranca.

Defina o ”Método de entrega para mail local:” para ”formato mbox em /var/mail”.

Seleccione ”Sim” para ”Dividir configuração em pequenos ficheiros?:”.

Crie entradas de palavra-passe para o smarthost ao editar ”`/etc/exim4/passwd.client`”.

```
$ sudo vim /etc/exim4/passwd.client
 ...
$ cat /etc/exim4/passwd.client
^smtp.*\.hostname\.dom:username@hostname.dom:password
```

Configure o exim4(8) com ”`QUEUERUNNER='queueonly'`”, ”`QUEUERUNNER='nodaemon'`”, etc. em ”`/etc/default/exi` para minimizar o uso de recursos do sistema. (opcional)

Inicie o `exim4` com o seguinte.

```
$ sudo systemctl start exim4
```

o nome de máquina em ”`/etc/exim4/passwd.client`” não deve ser o alias. Verifique o nome real da máquina com o seguinte.

```
$ host smtp.hostname.dom
smtp.hostname.dom is an alias for smtp99.hostname.dom.
smtp99.hostname.dom has address 123.234.123.89
```

Utilizo expressões regulares em ”`/etc/exim4/passwd.client`” para contornar o problema do alias. Provavelmente o SMTP AUTH funciona mesmo que o ISP mova a máquina apontada pelo alias.

Pode atualizar manualmente a configuração do `exim4` com o seguinte:

- atualizar os ficheiros de configuração do ”`exim4`” em ”`/etc/exim4/`”.

  - criar "`/etc/exim4/exim4.conf.localmacros`" para definir MACROs e editar "`/etc/exim4/exim4.conf.templ` (configuração não-dividida)
  - criar novos ficheiros ou editar ficheiros existentes nos sub-diretórios "`/etc/exim4/exim4.conf.d`". (configuração dividida)
- Execute "`systemctl reload exim4`".

Cuidado
O arranque do `exim4` demora muito tempo se foi escolhido "Não" (valor predefinido) na pergunta debconf de "Manter a quantidade de consultas DNS no mínimo (Chamar-a-pedido)?" e o sistema **não** estiver ligado à Internet durante o arranque.

Por favor leia o guia oficial em "`/usr/share/doc/exim4-base/README.Debian.gz`" e update-exim4.conf(8).

Atenção
Para qualquer consideração prática, utilize SMTP com STARTTLS na porta 587 ou SMTPS (SMTP sobre SSL) na porta 465, em vez de SMTP simples na porta 25.

6.2.4.2 A configuração do postfix com SASL

Para o mail de Internet através de smarthost, deve primeiro ler a documentação do postfix e páginas chave do manual.

comando	função
postfix(1)	Programa de controlo do postfix
postconf(1)	Utilitário de configuração do postfix
postconf(5)	Parâmetros de configuração do postfix
postmap(1)	Manutenção da tabela de buscas do Postfix
postalias(1)	Manutenção da base de dados de alias do Postfix

Tabela 6.4: Lista dos manuais importantes do postfix

(Re)configurar os pacotes `postfix` e `sasl2-bin` como a seguir.

```
$ sudo systemctl stop postfix
$ sudo dpkg-reconfigure postfix
```

Escolha "Internet com smarthost".

Defina "SMTP relay host (em branco para nenhum):" para "`[smtp.hostname.dom]:587`" e configure-o como o seguinte.

```
$ sudo postconf -e 'smtp_sender_dependent_authentication = yes'
$ sudo postconf -e 'smtp_sasl_auth_enable = yes'
$ sudo postconf -e 'smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd'
$ sudo postconf -e 'smtp_sasl_type = cyrus'
$ sudo vim /etc/postfix/sasl_passwd
```

Crie entradas de palavra-passe para o smarthost.

```
$ cat /etc/postfix/sasl_passwd
[smtp.hostname.dom]:587     username:password
$ sudo postmap hush:/etc/postfix/sasl_passwd
```

Arranque o `postfix` com o seguinte.

```
$ sudo systemctl start postfix
```

Aqui o uso de "`[`" e "`]`" no diálogo do `dpkg-reconfigure` e "`/etc/postfix/sasl_passwd`" assegura que não se verifica o registo MX mas usa directamente o nome de máquina exacto especificado. Veja "ativar autenticação SASL no cliente SMTP do Postfix" em "`/usr/share/doc/postfix/html/SASL_README.html`".

6.2.4.3 A configuração do endereço de mail

Existem alguns ficheiros de configuração de endereços para transporte, entrega e agentes de utilizador de mail.

ficheiro	função	aplicação
`/etc/mailname`	nome de máquina predefinido para mail (saída)	Específico de Debian, mailname(5)
`/etc/email-addresses`	nome de máquina para enganar o mail de saída	`ficheiros_de_configuração-exim4`(5) específicos do exim(8)
`/etc/postfix/generic`	nome de máquina para enganar o mail de saída	específico do postfix(1), activado após a execução do comando postmap(1).
`/etc/aliases`	alias de nome de conta para mail recebido	geral, activado após a execução do comando newaliases(1).

Tabela 6.5: Lista de ficheiros de configuração relacionados com endereços de mail

O **mailname** no ficheiro "`/etc/mailname`" é normalmente um nome de domínio totalmente qualificado (FQDN) que resolve para um dos endereços IP do anfitrião. Para a estação de trabalho móvel que não tem um nome de máquina com endereço IP resolvível, regule este **mailname** para o valor de "`hostname -f`". (Esta é uma escolha segura e funciona para ambos `exim4-*` e `postfix`.)

Dica
O conteúdo de "`/etc/mailname`" é utilizado por muitos programas não-MTA para o comportamento predefinido dele. Para o `mutt`, defina as variáveis "`hostname`" e "`from`" no ficheiro `~/muttrc` para sobrepor o valor **mailname**. Para programas no pacote `devscripts`, como o bts(1) e dch(1), exporte as variáveis de ambiente "`$DEBFULLNAME`" e "`$DEBEMAIL`" para o sobrepor.

Dica
O pacote `popularity-contest` normalmente envia mail a partir da conta de root com FQDN. Tem de definir `MAILFROM` em `/etc/popularity-contest.conf` como descrito no ficheiro `/usr/share/popularity-contest/default.conf`. Caso contrário, o seu mail será rejeitado pelo servidor SMTP do smarthost. Apesar de isto ser um tédio, esta aproximação é mais segura do que reescrever o endereço fonte para todos os mails do root pelo MTA e deve ser usado para outros daemons e scripts do cron.

Ao definir o **mailname** para "`hostname -f`", o spoofing do endereço de mail da fonte via MTA pode ser realizado com o seguinte.

- ficheiro "`/etc/email-addresses`" para exim4(8) conforme é explicado em exim4-config_files(5)
- ficheiro "`/etc/postfix/generic`" para postfix(1) conforme é explicado em generic(5)

Para o `postfix`, os seguintes passos adicionais são necessários:

```
# postmap hash:/etc/postfix/generic
# postconf -e 'smtp_generic_maps = hash:/etc/postfix/generic'
# postfix reload
```

Pode testar a configuração do endereço de email a usar o seguinte:

- exim(8) com as opções `-brw, -bf, -bF, -bV, …`
- postmap(1) com a opção `-q`.

Dica
O exim vem com vários programas utilitários como o exiqgrep(8) e exipick(8). Veja "`dpkg -L exim4-base|grep man8/`" para os comandos disponíveis.

6.2.4.4 Operações MTA básicas

Existem várias operações MTA básicas. Algumas podem ser executadas através do interface de compatibilidade do sendmail(1).

comando exim	comando postfix	descrição
`sendmail`	`sendmail`	lê mails da entrada standard e prepara a entrega (`-bm`)
`mailq`	`mailq`	lista a lista de espera de mail com estado e ID de lista de espera (`-bp`)
`newaliases`	`newaliases`	inicializa a base de dados e alias (`-I`)
`exim4 -q`	`postqueue -f`	enxagua mails em espera (`-q`)
`exim4 -qf`	`postsuper -r ALL deferred; postqueue -f`	enxagua todos mails
`exim4 -qff`	`postsuper -r ALL; postqueue -f`	enxagua até mails congelados
`exim4 -Mg queue_id`	`postsuper -h queue_id`	congela uma mensagem pelo seu ID de lista de espera
`exim4 -Mrm queue_id`	`postsuper -d queue_id`	remove uma mensagem pelo seu ID de lista de espera
N/D	`postsuper -d ALL`	remove todas as mensagens

Tabela 6.6: Lista de operações MTA básicas

Dica
Poderá ser uma boa ideia enxaguar todos os mails por um script em "`/etc/ppp/ip-up.d/*`".

6.3 O servidor de acesso remoto e utilitários (SSH)

O Secure SHell (SSH) é o modo **seguro** de efectuar ligações na Internet. Uma versão livre do SSH chamada OpenSSH está disponível nos pacotes `openssh-client` e `openssh-server` em Debian.

Para o utilizador as funções do ssh(1) são uma telnet(1) mais inteligente e segura. Ao contrário do comando `telnet`, o comando `ssh` não pára no caractere de escape do `telnet` (predefinição inicial CTRL-]).

Embora o `shellinabox` não seja um programa SSH, ele é listado aqui como uma alternativa interessante para o acesso remoto ao terminal.

Veja também Secção 7.9 para conectar a programas clientes X remotos.

pacote	popcon	tamanho	ferramenta	descrição
openssh-client	V:682, I:997	3611	ssh(1)	Cliente de shell segura
openssh-server	V:774, I:820	3615	sshd(8)	Servidor de shell segura
ssh-askpass	I:15	103	ssh-askpass(1)	pede ao utilizador uma frase passe para ssh-add (X simples)
ssh-askpass-gnome	V:0.5, I:2.9	46	ssh-askpass-gnome(1)	pede ao utilizador uma frase passe para ssh-add (GNOME)
ssh-askpass-fullscreen	V:0.09, I:0.40	41	ssh-askpass-fullscreen(1)	pede ao utilizador uma frase passe para ssh-add (GNOME) que seja atrativa
shellinabox	V:0.61, I:0.94	528	shellinaboxd(1)	servidor web para emulador de terminal VT100 acessível por browser

Tabela 6.7: Lista de servidores de acesso remoto e utilitários



Cuidado
Veja Secção 4.6.3 se o seu SSH for acessível a partir da Internet.

Dica
Por favor use o programa screen(1) para ativar a sobrevivência do processo de shell remota à interrupção da ligação (veja Secção 9.1.2).

6.3.1 Bases do SSH

O daemon SSH OpenSSH suporta apenas o protocolo SSH 2.

Por favor leia "`/usr/share/doc/openssh-client/README.Debian.gz`", ssh(1), sshd(8), ssh-keygen(1), ssh-add(1) e ssh-agent(1).



Atenção
o "`/etc/ssh/sshd_not_to_be_run`" não pode estar presente se desejar correr o servidor OpenSSH. Não ative a autenticação baseada em rhost (`HostbasedAuthentication` em `/etc/ssh/sshd_config`).

ficheiro de configuração	descrição do ficheiro de configuração
`/etc/ssh/ssh_config`	Predefinições do cliente SSH, veja ssh_config(5)
`/etc/ssh/sshd_config`	Predefinições do servidor SSH, veja sshd_config(5)
`~/.ssh/authorized_keys`	chaves SSH públicas predefinidas que os clientes usam para ligar a esta conta neste servidor SSH
`~/.ssh/id_rsa`	chave SSH-2 RSA secreta do utilizador
`~/.ssh/id_`*`key-type-name`*	chave secreta SSH-2 *tipo-nome*, como `ecdsa`, `ed25519`, ... do utilizador

Tabela 6.8: Lista de ficheiros de configuração do SSH

O seguinte inicia uma ligação ssh(1) a partir de um cliente.

comando	descrição
`ssh username@hostname.domain.ext`	ligar com modo predefinido
`ssh -v username@hostname.domain.ext`	ligar com modo predefinido com mensagens de depuração
`ssh -o PreferredAuthentications=password username@hostname.domain.ext`	força o uso de palavra-passe com SSH versão 2
`ssh -t username@hostname.domain.ext passwd`	execute o programa `passwd` para atualizar a palavra-passe num anfitrião remoto

Tabela 6.9: Lista de exemplos de arranque do cliente SSH

6.3.2 Nome de utilizador no anfitrião remoto

Se utilizar o mesmo nome de utilizador no anfitrião local e no anfitrião remoto, pode deixar de escrever "`username@`".

Mesmo que utilize um nome de utilizador diferente no anfitrião local e remoto, pode eliminá-lo utilizando "`~/.ssh/config`". Para o serviço Debian Salsa com o nome de conta "`foo-guest`", você define "`~/.ssh/config`" para conter o seguinte.

```
Host salsa.debian.org people.debian.org
User foo-guest
```

6.3.3 Ligar sem palavras-passe remotas

É possível evitar ter de se lembrar de palavras-passe para sistemas remotos utilizando "`PubkeyAuthentication`" (protocolo SSH-2).

No sistema remoto, defina as respetivas entradas, "`PubkeyAuthentication yes`", em "`/etc/ssh/sshd_config`".

Crie chaves de autenticação localmente e instale a chave pública no sistema remoto como o seguinte.

```
$ ssh-keygen -t rsa
$ cat .ssh/id_rsa.pub | ssh user1@remote "cat - >>.ssh/authorized_keys"
```

Pode adicionar opções às entradas em "`~/.ssh/authorized_keys`" para limitar máquinas e correr comandos específicos. Veja sshd(8) "AUTHORIZED_KEYS FILE FORMAT".

6.3.4 Lidar com clientes SSH alienígenas

Existem alguns clientes SSH livres disponíveis para outras plataformas.

ambiente	programa de SSH livre
Windows	puTTY (PuTTY: um cliente SSH e Telnet gratuito) (GPL)
Windows (cygwin)	SSH no cygwin (Cygwin: Tenha aquela sensação de Linux - no Windows) (GPL)
Mac OS X	OpenSSH; use `ssh` na aplicação de Terminal (GPL)

Tabela 6.10: Lista de clientes SSH para outras plataformas

6.3.5 Configurar o ssh-agent

É mais seguro proteger as suas chaves secretas de autenticação SSH com uma frase-passe. Se nenhuma frase-passe foi definida, use "`ssh-keygen -p`" para a definir.

Ponha a sua chave SSH pública (ex. "`~/.ssh/id_rsa.pub`") em "`~/.ssh/authorized_keys`" numa máquina remota a usar uma ligação à máquina remota baseada em palavra-passe como descrito em cima.

```
$ ssh-agent bash
$ ssh-add ~/.ssh/id_rsa
Enter passphrase for /home/username/.ssh/id_rsa:
Identity added: /home/username/.ssh/id_rsa (/home/username/.ssh/id_rsa)
```

Nenhuma palavra-passe necessária desde aqui até ao próximo comando.

```
$ scp foo username@remote.host:foo
```

Carregue em ^D (CTRL-D) para terminar a sessão do ssh-agent.

Para o servidor X, o script de arranque normal de Debian executa o `ssh-agent` como o processo pai. Portanto apenas precisa de executar o `ssh-add` uma vez. Para mais, leia ssh-agent(1) e ssh-add(1).

6.3.6 Enviar uma mensagem de correio eletrónico a partir de um anfitrião remoto

Se tiver uma conta de shell SSH num servidor com definições de DNS adequadas, pode enviar um e-mail gerado na sua estação de trabalho como um e-mail genuinamente enviado do servidor remoto.

```
$ ssh username@example.org /usr/sbin/sendmail -bm -ti -f "username@example.org" < mail_data ↩
    .txt
```

6.3.7 Reencaminhamento de portos para SMTP/POP3 em túnel

Para estabelecer um pipe para ligação ao porto 25 do `servidor remoto` a partir do porto 4025 da `máquina local`, e para a porta 110 do `servidor remoto` a partir do porto 4110 da `máquina local` através de `ssh`, execute na máquina local como a seguir.

```
# ssh -q -L 4025:remote-server:25 4110:remote-server:110 username@remote-server
```

Este é um modo seguro de fazer ligações a servidores SMTP/POP3 pela Internet. Configure a entrada "`AllowTcpForwardi`
para "`yes`" em "`/etc/ssh/sshd_config`" na máquina remota.

6.3.8 Como desligar o sistema remoto em SSH

Precisa de proteger o processo ao fazer "`shutdown -h now`" (veja Secção 1.1.8) a partir da terminação do SSH a usar o comando at(1) (veja Secção 9.4.13) com o seguinte.

```
# echo "shutdown -h now" | at now
```

Correr "`shutdown -h now`" numa sessão do screen(1) (veja Secção 9.1.2) é outro modo de fazer o mesmo.

6.3.9 Depurar problemas no SSH

Se estiver com problemas, verifique as permissões dos ficheiros de configuração e corra o `ssh` com a opção "`-v`".

Use a opção "`-p`" se for o root e esteja a ter problemas com a firewall; isto evita o uso dos portos de servidor 1 -- 1023.

Se as ligações `ssh` a um site remoto subitamente deixarem de funcionar, pode ser o resultado de reparações pelo administrador do sistema, mais provável uma alteração na "`host_key`" durante a manutenção do sistema. Após certificar-se que é este o caso e ninguém está a tentar falsificar a máquina remota com algum hack inteligente, pode-se recuperar a ligação ao remover a entrada "`host_key`" de "`~/.ssh/known_hosts`" na máquina local.

6.4 O servidor de impressão e utilitários

No antigo sistema do tipo Unix, o daemon de impressão em linha (lpd) BSD era o padrão e o formato de impressão padrão do software livre clássico era o PostScript (PS). Era utilizado um sistema de filtros juntamente com o Ghostscript para permitir a impressão numa impressora não-PostScript. Ver Secção 11.4.1.

No sistema Debian moderno, o Common UNIX Printing System (CUPS) é de facto o padrão e o formato padrão obtido na saída é o livre e moderno Portable Document Format (PDF).

O CUPS utiliza o Protocolo de Impressão Internet (IPP). O IPP é o standard de facto multiplataforma para impressão remota com capacidade de comunicação bidirecional.

Graças à funcionalidade de auto-conversão dependente do formato de ficheiro do sistema CUPS, simplesmente fornecer quaisquer dados ao comando `lpr` deverá gerar a saída de impressão esperada. (No CUPS, o `lpr` pode ser activado ao instalar o pacote `cups-bsd`.)

O sistema Debian tem alguns pacotes notáveis para os servidores e utilitários de impressão.

pacote	popcon	tamanho	porto	descrição
`lpr`	V:1.9, I:2.2	378	impressora (515)	BSD lpr/lpd (daemon de impressora de linha)
`cups`	V:144, I:447	1088	IPP (631)	Servidor CUPS de Impressão em Internet
`cups-client`	V:152, I:459	429	, ,	Comandos de impressão do System V para o CUPS: lp(1), lpstat(1), lpoptions(1), cancel(1), lpmove(8), lpinfo(8), lpadmin(8), …
`cups-bsd`	V:34, I:182	131	, ,	comandos de impressão BSD para o CUPS: lpr(1), lpq(1), lprm(1), lpc(8)
`printer-driver-gutenprint`	V:11, I:52	1117	Não aplicável	Drivers de impressoras para o CUPS

Tabela 6.11: Lista de servidores de impressoras e utilitários

Dica
Pode configurar o sistema CUPS ao apontar o seu explorador web para "http://localhost:631/" .

6.5 Outras aplicações de servidor de rede

Aqui estão outras aplicações de servidor de rede.

Common Internet File System Protocol (CIFS) é o mesmo protocolo que Server Message Block (SMB) e é bastante usado pelo Microsoft Windows.

pacote	popcon	tamanho	protocolo	descrição
`telnetd`	V:0.3, I:1.6	56	TELNET	Servidor TELNET
`nfs-kernel-server`	V:47, I:54	874	NFS	Partilha de ficheiros do Unix
`samba`	V:110, I:123	5182	SMB	Partilha de ficheiros e impressoras do Windows
`netatalk`	V:0.62, I:0.85	933	ATP	Partilha de ficheiros e impressoras do Apple/Mac (AppleTalk)
`proftpd-basic`	V:3.3, I:8.4	452	FTP	Descarrega de ficheiros geral
`apache2`	V:175, I:211	587	HTTP	Servidor web geral
`squid`	V:9.0, I:9.4	9583	, ,	servidor proxy web geral
`bind9`	V:33, I:35	904	DNS	Endereço IP para outras máquinas
`kea`	I:0.51	245	DHCP	Endereço IP do próprio cliente

Tabela 6.12: Lista de outras aplicações de servidor de rede

Dica
Veja Secção 4.5.2 para integração de sistemas servidor.

Dica
A resolução do nome de máquinas é normalmente disponibilizada pelo servidor de DNS. Para o endereço IP atribuído dinamicamente à máquina por DHCP, pode ser definido DNS Dinâmico para a resolução do nome de máquina a utilizar o `bind9` e o `kea` conforme é descrito na página DDNS no wiki Debian.

Dica
O uso de servidor proxy como o `squid` é muito mais eficiente para poupar largura de banda que o uso de servidor mirror local com o conteúdo completo do arquivo Debian.

6.6 Outros clientes de aplicação de rede

Aqui estão outros clientes de aplicação de rede.

6.7 Os diagnósticos dos daemons do sistema

O programa `telnet` activa ligação manual aos daemons do sistema e aos seus diagnósticos.

Para testar o serviço POP3 simples, tente o seguinte:

```
$ telnet mail.ispname.net pop3
```

Para testar o serviço POP3 com TLS/SSL ativo por alguns ISPs, precisa do cliente `telnet` com TLS/SSL ativado pelos pacotes `telnet-ssl` ou `openssl`.

```
$ telnet -z ssl pop.gmail.com 995
```

```
$ openssl s_client -connect pop.gmail.com:995
```

Os seguintes RFCs disponibilizam o conhecimento necessário para cada daemon de sistema.

A utilização de portos é descrita em "`/etc/services`".

pacote	popcon	tamanho	protocolo	descrição
netcat-traditional	V:55, I:906	139	TCP/IP	Canivete suíço do TCP/IP
netcat-openbsd	V:22, I:119	105	TCP/IP	Canivete suíço do TCP/IP com suporte para IPv6, proxies e sockets Unix
openssl	V:846, I:996	2564	SSL	Binário Secure Socket Layer (SSL) e ferramentas criptográficas relacionadas
stunnel4	V:6.6, I:9.9	24	, ,	revestimento SSL universal
telnet	V:10, I:190	56	TELNET	Cliente TELNET
nfs-common	V:147, I:191	1158	NFS	Partilha de ficheiros do Unix
smbclient	V:25, I:206	2117	SMB	Cliente de partilha de ficheiros e impressoras do MS Windows
cifs-utils	V:32, I:118	368	, ,	Comandos de montar e desmontar para ficheiros remotos do MS Windows
wget	V:181, I:982	3784	HTTP e FTP	descarregador de web
curl	V:260, I:727	520	, ,	, ,
transmission-gtk	V:9, I:143	6259	BitTorrent	Cliente BitTorrent (GTK)
transmission-qt	V:0.6, I:2.6	6213	, ,	Cliente do BitTorrent (Qt)
ktorrent	V:1.3, I:5.0	5031	, ,	Cliente do BitTorrent (Qt)
qbittorrent	V:8, I:23	15378	, ,	Cliente do BitTorrent (Qt)
bind9-host	V:124, I:940	136	DNS	host(1) do bind9, ”Prioridade: standard”
dnsutils	V:5, I:146	23	, ,	dig(1) do bind, ”Prioridade: standard”
ldap-utils	V:11, I:57	790	LDAP	obter dados de um servidor LDAP

Tabela 6.13: Lista de clientes de aplicação de rede

RFC	descrição
rfc1939 e rfc2449	serviço POP3
rfc3501	serviço IMAP4
rfc2821 (rfc821)	serviço SMTP
rfc2822 (rfc822)	Formato de ficheiro de mail
rfc2045	Extensões de correio de Internet multiuso(MIME)
rfc819	serviço DNS
rfc2616	serviço HTTP
rfc2396	definição URI

Tabela 6.14: Lista de RFCs populares

Capítulo 7

Sistema GUI (interface gráfica de utilizador)

7.1 Ambiente de trabalho GUI

Existem várias escolhas para o ambiente de trabalho GUI completo no sistema Debian.

pacote de tarefas	popcon	tamanho	descrição
task-gnome-desktop	I:192	9	Ambiente de trabalho GNOME
task-xfce-desktop	I:88	9	Ambiente de trabalho Xfce
task-kde-desktop	I:94	6	Ambiente de trabalho KDE Plasma
task-mate-desktop	I:30	9	Ambiente de trabalho MATE
task-cinnamon-desktop	I:35	9	Ambiente de trabalho Cinnamon
task-lxde-desktop	I:19	9	Ambiente de trabalho LXDE
task-lxqt-desktop	I:16	9	Ambiente de trabalho LXQt
task-gnome-flashback-desktop	I:9.6	6	Ambiente de trabalho GNOME Flashback

Tabela 7.1: Lista de ambientes de trabalho

Dica
Os pacotes de dependências selecionados por um meta-pacote de tarefas podem estar fora de sincronia com o último estado de transição de pacotes sob o ambiente Debian `unstable`/`testing`. Para `task-gnome-desktop`, pode ser necessário ajustar as seleções de pacotes como se segue:

- Inicie aptitude(8) como `sudo aptitude -u`.
- Mova o cursor para ”Tarefas” e prima ”Enter”.
- Mova o cursor para ”Utilizador final” e prima ”Enter”.
- Mova o cursor para ”GNOME” e prima ”Enter”.
- Mova o cursor para `task-gnome-desktop` e prima ”Enter”.
- Mova o cursor para ”Dependências” e prima ”m” (selecionado manualmente).
- Mova o cursor para ”Recomendado” e prima ”m” (seleção manual).
- Mova o cursor para ”`task-gnome-desktop` ” e prima ”-”. (excluir)
- Ajusta os pacotes selecionados e elimina os pacotes problemáticos que causam conflitos de pacotes.
- Prima ”g” para iniciar a instalação.

Este capítulo irá focar-se principalmente no ambiente de trabalho predefinido de Debian: `task-gnome-desktop` que oferece o GNOME em wayland.

7.2 Protocolo de comunicação GUI

O protocolo de comunicação GUI utilizado no ambiente de trabalho GNOME pode ser:

- Wayland (protocolo de servidor de ecrã) (nativo)
- Protocolo central do sistema X Window (via `xwayland`)

Consulte o site freedesktop.org para saber como a arquitetura Wayland é diferente da arquitetura X Window.

Do ponto de vista do utilizador, as diferenças podem ser resumidas coloquialmente como:

- Wayland é um protocolo de comunicação GUI para o mesmo host: novo, mais simples, mais rápido, sem binário raiz setuid
- X Window é um protocolo de comunicação GUI com capacidade de rede: tradicional, complexo, mais lento, binário de raiz setuid

Para as aplicações que utilizam o protocolo Wayland, o acesso ao seu conteúdo de visualização a partir de um anfitrião remoto é suportado pelo VNC ou RDP. Ver Secção 7.8

Os servidores X modernos têm a Extensão de Memória Partilhada do MIT e comunicam com os seus clientes X locais utilizando a memória partilhada local. Isso contorna o canal de comunicação interprocessos Xlib transparente da rede e ganha desempenho. Esta situação foi o antecedente da criação do Wayland como um protocolo de comunicação GUI apenas local.

Utilizando o programa `xeyes` iniciado a partir do terminal GNOME, pode verificar o protocolo de comunicação GUI utilizado por cada aplicação GUI.

```
$ xeyes
```

- Se o cursor do rato estiver numa aplicação como o ”terminal GNOME”, que utiliza o protocolo de servidor de visualização Wayland, os olhos não se movem com o cursor do rato.
- Se o cursor do rato estiver numa aplicação como o ”xterm”, que utiliza o protocolo central do sistema X Window, os olhos movem-se com o cursor do rato, expondo a natureza não tão isolada da arquitetura do X Window.

A partir de abril de 2021, muitas aplicações GUI populares, como as aplicações de GNOME e LibreOffice (LO), foram migradas para o protocolo de servidor de ecrã Wayland. Vejo que as aplicações `xterm`, `gitk`, `chromium`, `firefox`, `gimp`, `dia`, e KDE ainda utilizam o protocolo central do X Window System.

Nota
Tanto para o xwayland no Wayland como para o X Window System nativo, o antigo ficheiro de configuração do servidor X ”`/etc/X11/xorg.conf`” não deve existir no sistema. Os dispositivos gráficos e de entrada são agora configurados pelo kernel com DRM, KMS, e udev. O servidor X nativo foi reescrito para os utilizar. Veja ”suporte a modo de vídeo predefinido modedb” na documentação do kernel Linux.

7.3 Infraestrutura GUI

Aqui estão os pacotes de infraestrutura GUI notáveis para o ambiente GNOME em Wayland.

pacote	popcon	tamanho do pacote	descrição
`mutter`	V:1, I:22	216	O gerenciador de janelas mutter do GNOME **[auto]**
`xwayland`	V:256, I:335	2652	Um servidor X a correr em cima do wayland **[auto]**
`gnome-remote-desktop`	V:147, I:241	2743	Daemon de desktop remoto para GNOME usando PipeWire **[auto]**
`gnome-tweaks`	V:17, I:232	1145	Definições de configuração avançadas para o GNOME
`gnome-shell-extension-prefs`	V:8, I:117	73	Ferramenta para ativar / desativar extensões do GNOME Shell

Tabela 7.2: Lista de pacotes de infra-estruturas GUI notáveis

Aqui, ”**[auto]**” significa que estes pacotes são instalados automaticamente quando `task-gnome-desktop` é instalado.

Dica
O `gnome-tweaks` é o utilitário de configuração indispensável. Por exemplo:

- Pode forçar a ”Sobre-amplificação” do volume do som a partir de ”Geral”.
- Pode forçar ”Caps” a tornar-se ”Esc” em ”Teclado & Rato” -> ”Teclado” -> ”Opção de esquema adicional”.

Dica
As caraterísticas detalhadas do ambiente de trabalho GNOME podem ser configuradas com utilitários iniciados escrevendo ”`settings`”, ”`tweaks`”, ou ”`extensions`” depois de premir a tecla `Super-`.

7.4 Aplicações GUI

Muitas aplicações GUI úteis estão agora disponíveis em Debian. Instalar pacotes de software como o `scribus` (KDE) no ambiente de desktop GNOME é bastante aceitável dado que a funcionalidade correspondente não está disponível no ambiente de desktop GNOME. Mas a instalação de demasiados pacotes com funcionalidades duplicadas pode sobrecarregar o seu sistema.

Aqui está uma lista de aplicações que me chamaram a atenção.

7.5 Diretórios de utilizador

Os nomes predefinidos para as diretorias de utilizador, tais como "`~/Desktop`", "`~/Documents`", ..., utilizados pelo ambiente de trabalho dependem da configuração regional utilizada para a instalação do sistema. Pode redefini-los para os nomes ingleses através de:

```
$ LANGUAGE=C xdg-user-dirs-update --force
```

Depois mova manualmente todos os dados para os diretórios mais recentes. Veja xdg-user-dirs-update(1).

Pode também defini-los com qualquer nome editando "`~/.config/user-dirs.dirs`". Veja user-dirs.dirs(5).

7.6 Fontes

Muitas fontes escaláveis úteis estão disponíveis para os utilizadores em Debian. A preocupação do utilizador é como evitar a redundância e como configurar partes das fontes instaladas para serem desativadas. Caso contrário, as opções de tipos de letra inúteis podem sobrecarregar os menus das aplicações GUI.

O sistema Debian usa a biblioteca FreeType 2.0 para rasterizar muitos formatos de fontes escaláveis para ecrã e impressão:

- tipos de letra Tipo 1 (PostScript) que usam curvas de Bézier cúbicas (formato quase obsoleto)
- Tipos de letra TrueType que utilizam curvas Bézier quadráticas (formato de boa escolha)
- Tipos de letra OpenType que utilizam curvas Bézier cúbicas (formato de melhor escolha)

7.6.1 Fontes (tipos de letra) básicas

A tabela que se segue foi compilada na esperança de ajudar os utilizadores a escolherem fontes escaláveis adequadas com uma compreensão clara da compatibilidade métrica e da cobertura de glifos. A maioria das fontes cobre todos os caracteres latinos, gregos e cirílicos. A escolha final das fontes ativadas também pode ser afetada pela sua estética. Estas fontes podem ser utilizadas para a visualização no ecrã ou para a impressão em papel.

Aqui:

- "MCM" significa "métrica compatível com os tipos de letra fornecidos pela Microsoft"
- "MCMATC" significa "métrica compatível com os tipos de letra fornecidos pela Microsoft: Arial, Times New Roman, Courier New"
- "MCAHTC" significa "métrica compatível com os tipos de letra fornecidos pela Adobe: Helvética, Times, Courier"
- Os números nas colunas de tipo de letra representam a largura "M" relativa aproximada para o mesmo tamanho de letra.

pacote	popcon	tamanho do pacote	tipo	descrição
evolution	V:27, I:229	489	GNOME	Gestor de Informações Pessoais (groupware e email)
thunderbird	V:39, I:102	303329	GTK	Cliente de correio eletrónico (Mozilla Thunderbird)
kontact	V:1, I:10	2360	KDE	Gestor de Informações Pessoais (groupware e email)
libreoffice-writer	V:105, I:422	34232	LO	processador de texto
abiword	V:0.8, I:4.7	3596	GNOME	processador de texto
calligrawords	V:0.2, I:2.7	7144	KDE	processador de texto
scribus	V:1, I:12	32942	KDE	desktop publishing editor para editar ficheiros PDF
glabels	V:0.3, I:2.5	1283	GNOME	editor de etiquetas
libreoffice-calc	V:101, I:419	29620	LO	folha de cálculo
gnumeric	V:3, I:10	9945	GNOME	folha de cálculo
calligrasheets	V:0.1, I:1.7	14014	KDE	folha de cálculo
libreoffice-impress	V:88, I:417	2499	LO	apresentação
calligrastage	V:0.1, I:1.6	6237	KDE	apresentação
libreoffice-base	V:21, I:67	5130	LO	gestão de base de dados
kexi	V:0.05, I:0.80	7565	KDE	gestão de base de dados
libreoffice-draw	V:89, I:418	11418	LO	editor de gráficos vectoriais (desenho)
inkscape	V:12, I:76	113353	GNOME	editor de gráficos vectoriais (desenho)
karbon	V:0.1, I:2.1	4029	KDE	editor de gráficos vectoriais (desenho)
dia	V:1, I:16	3846	GTK	editor de fluxogramas e diagramas
gimp	V:34, I:211	33315	GTK	editor de gráficos bitmap (pintura)
shotwell	V:14, I:246	6334	GTK	organizador de fotos digitais
digikam	V:1.5, I:8.3	333	KDE	organizador de fotos digitais
darktable	V:4, I:11	36092	GTK	mesa de luz e câmara escura para fotógrafos
gnome-video-trimmer	V:0.1, I:1.0	1665	GNOME	corte sem perdas de ficheiros de vídeo
kdenlive	V:3, I:21	22134	KDE	editor de vídeo não-linear
shotcut	V:1.0, I:7.5	8270	Aplicação Qt	editor de vídeo não-linear
openshot-qt	V:0.9, I:8.8	196007	Aplicação Qt	editor de vídeo não-linear
flowblade	V:0.2, I:2.2	42213	GTK	editor de vídeo não-linear
planner	V:0.2, I:4.2	1400	GNOME	gestão de projectos
calligraplan	V:0.2, I:2.9	23545	KDE	gestão de projectos
gnucash	V:2.0, I:6.7	29434	GNOME	contas pessoais
homebank	V:0.3, I:1.5	3194	GTK	contas pessoais
kmymoney	V:0.4, I:1.9	19297	KDE	contas pessoais
frescobaldi	V:0.3, I:1.5	11102	Aplicação Qt	editor de texto de partituras musicais LilyPond
librecad	V:1, I:13	9164	Aplicação Qt	sistema de desenho assistido por computador (CAD) (2D)
freecad	V:1, I:20	112	Aplicação Qt	sistema de conceção assistida por computador (CAD) (3D)
kicad	V:3, I:14	224155	GTK	software de desenho de esquemas electrónicos e PCB
xsane	V:9, I:128	1512	GTK	frontend para digitalizador (scanner)
libreoffice-math	V:84, I:420	1982	LO	editor de fórmulas/equações matemáticas
calibre	V:7, I:24	71216	KDE	conversor de e-books e gestor de biblioteca
fbreader	V:0.7, I:5.4	3827	GTK	leitor de e-book
evince	V:76, I:284	953	GNOME	visualizador de documentos (pdf)
okular	V:42, I:130	4544	KDE	visualizador de documentos (pdf)

pacote	popcon	tamanho	sans	serif	mono	nota sobre o tipo de letra
fonts-cantarell	V:174, I:288	224	59	-	-	Cantarell (GNOME 3, ecrã)
fonts-noto	I:150	30	61	63	40	Noto fonts (Google, multilingue com CJK)
fonts-dejavu	I:385	34	58	68	40	DejaVu (GNOME 2, MCM:Verdana, estendida Bitstream Vera)
fonts-liberation2	V:40, I:156	15	56	60	40	Liberation fonts para LibreOffice (Red Hat, MCMATC)
fonts-croscore	V:20, I:36	5260	56	60	40	Chrome OS: Arimo, Tinos e Cousine (Google, MCMATC)
fonts-crosextra-carlito	V:20, I:90	2696	57	-	-	Chrome OS: Carlito (Google, MCM:Calibri)
fonts-crosextra-caladea	V:12, I:83	347	-	55	-	Chrome OS: Caladea (Google, MCM:Cambria) (Latin only)
fonts-freefont-ttf	V:78, I:203	14460	57	59	40	GNU FreeFont (estendida URW Nimbus)
fonts-quicksand	V:210, I:452	392	56	-	-	Debian task-desktop, Quicksand (ecrã, apenas Latin)
fonts-hack	V:34, I:136	2509	-	-	40 P	Um tipo de letra concebido para o código fonte Hack (Facebook)
fonts-sil-gentiumplus	V:0, I:28	14345	-	54	-	Gentium SIL
fonts-sil-charis	V:1, I:28	6704	-	59	-	Charis SIL
fonts-urw-base35	V:187, I:534	15558	56	60	40	URW Nimbus (Nimbus Sans, Roman No. 9 L, Mono L, MCAHTC)
fonts-ubuntu	V:2.2, I:4.7	4339	58	-	33 P	Ubuntu fonts (ecrã)
fonts-terminus	V:0.3, I:3.9	451	-	-	33	Cool retro terminal fonts
ttf-mscorefonts-installer	V:1, I:40	85	56?	60	40	Descarregador de tipos de letra não gratuitos da Microsoft (ver abaixo)

Tabela 7.4: Lista de tipos de letra notáveis TrueType e OpenType

- O ”P” em colunas de tipo de letra mono representa a sua facilidade de utilização para a programação, com ”0”/”O” e ”1”/”I”/”l” claramente distinguíveis.
- O pacote `ttf-mscorefonts-installer` descarrega as ”Fonts de núcleo para a Web” da Microsoft e instala Arial, Times New Roman, Courier New, Verdana, ... Estes dados de tipos de letra instalados são dados não livres.

Muitos tipos de letra latinos gratuitos têm a sua linhagem na família URW Nimbus ou Bitstream Vera.

Dica
Se a sua localidade precisa de tipos de letra que não estão bem cobertos pelos tipos de letra acima, por favor utilize o aptitude para verificar os pacotes de tarefas listados em ”Tarefas” -> ”Localização”. Os pacotes de fontes listados como ”Dependências:” ou ”Recomendados:” nos pacotes de tarefas de localização são os principais candidatos.

7.6.2 Rasterização de tipos de letra

Debian usa FreeType para rasterizar as fontes. A sua infraestrutura de escolha de fontes é fornecida pela biblioteca de configuração de fontes Fontconfig.

pacote	popcon	tamanho	descrição
`libfreetype6`	V:589, I:997	1050	FreeType biblioteca de rasterização de tipos de letra
`libfontconfig1`	V:577, I:808	344	Fontconfig, uma biblioteca de configuração de tipos de letra genérica -- binários de suporte
`fontconfig`	V:471, I:693	415	`fc-*`: comandos CLI para Fontconfig
`font-manager`	V:2.3, I:6.6	1116	Gestor de Tipos de Letra: comando para GUI Fontconfig
`nautilus-font-manager`	V:0.07, I:0.38	38	Extensão Nautilus para o Gestor de Tipos de Letra

Tabela 7.5: Lista de fontes ambiente notáveis e pacotes relacionados

Dica
Alguns pacotes de tipos de letra, como o `fonts-noto*`, instalam demasiados tipos de letra. Poderá também querer manter alguns pacotes de tipos de letra instalados mas desactivados numa situação de utilização normal. Os glifos múltiplos são esperados para alguns pontos de código Unicode devido à unificação Han e os glifos indesejados podem ser escolhidos pela biblioteca Fontconfig não configurada. Um dos casos mais incómodos é ”U+3001 IDEOGRAPHIC COMMA” e ”U+3002 IDEOGRAPHIC FULL STOP” entre os países CJK. Pode evitar facilmente esta situação problemática configurando a disponibilidade dos tipos de letra utilizando a GUI do gestor de tipos de (letra (font-manager).

Pode verificar informação de configuração de tipos de letra pelo seguinte.

- ”`fc-match(1)`” para o tipo de letra predefinido do fontconfig
- ”`fc-list(1)`” para tipos de letra disponíveis fontconfig

Pode configurar o estado de configuração da fonte a partir do editor de texto mas isto não é trivial. Veja fonts.conf(5).

7.7 Sandbox

Muitas das aplicações GUI em Linux estão disponíveis em formatos binários a partir de fontes não-Debian.

- AppImage -- Aplicações Linux que correm em qualquer lugar
- FLATHUB -- Aplicações para Linux, aqui mesmo
- snapcraft -- A loja de aplicações para Linux



Atenção
Os binários destes sítios podem incluir pacotes de software proprietário não-livre.

Existe alguma razão de ser para estas distribuições em formato binário para os aficionados do Software Livre que usam Debian, uma vez que estas podem acomodar um conjunto limpo de bibliotecas usadas para cada aplicação pelo respetivo programador original, independentemente das fornecidas por Debian.

O risco inerente de executar binários externos pode ser reduzido usando o ambiente sandbox que aproveita os recursos modernos de segurança do Linux (veja Secção 4.7.5).

- Para binários do AppImage e de alguns sites upstream, execute-os no firejail com configuração manual.
- Para binários do FLATHUB, execute-os no Flatpak . (Não é necessária nenhuma configuração manual.)
- Para binários do snapcraft, execute-os no Snap . (Não é necessária configuração manual. Compatível com programas daemon.)

O pacote `xdg-desktop-portal` fornece uma API padronizada para recursos comuns de desktop. Veja xdg-desktop-portal (flatpak) e xdg-desktop-portal (snap) .

pacote	popcon	tamanho	descrição
`flatpak`	V:104, I:115	9293	Estrutura de implantação de aplicações Flatpak para aplicações de ambiente de trabalho
`gnome-software-plugin-flatpak`	V:29, I:41	291	Suporte Flatpak para o software GNOME
`snapd`	V:58, I:62	62738	Daemon e ferramentas que habilitam pacotes snap
`gnome-software-plugin-snap`	V:1.4, I:2.2	144	Suporte Snap para o software GNOME
`xdg-desktop-portal`	V:373, I:439	2291	portal de integração no ambiente de trabalho para Flatpak e Snap
`xdg-desktop-portal-gtk`	V:337, I:438	715	backend xdg-desktop-portal para gtk (GNOME)
`xdg-desktop-portal-kde`	V:81, I:112	3333	backend do xdg-desktop-portal para Qt (KDE)
`xdg-desktop-portal-wlr`	V:2.4, I:6.6	159	backend do xdg-desktop-portal para wlroots (Wayland)
`firejail`	V:1.2, I:4.4	1827	um programa de segurança sandbox SUID firejail para uso com o AppImage

Tabela 7.6: Lista de ambientes sandbox notáveis e pacotes relacionados

Esta tecnologia de espaço isolado ”sandbox” é muito semelhante às aplicações no sistema operativo dos telefones inteligentes, em que as aplicações são executadas com acessos controlados aos recursos.

Algumas aplicações GUI grandes tais como navegadores web em Debian também usam a tecnologia de ambiente sandbox internamente para torná-los mais seguros.

7.8 Área de trabalho remota

pacote	popcon	tamanho	protocolos	descrição
gnome-remote-desktop	V:14, I:241	2743	RDP	Servidor de Ambiente de Trabalho Remoto GNOME
xrdp	V:28, I:31	4757	RDP	xrdp, servidor de Protocolo de Ambiente de Trabalho Remoto (RDP)
x11vnc	V:9, I:46	1863	RFB (VNC)	x11vnc, servidor VNC (Remote Framebuffer Protocol)
tigervnc-standalone-server	V:5, I:15	3124	RFB (VNC)	TigerVNC, servidor VNC (Remote Framebuffer Protocol)
gnome-connections	V:6, I:149	1726	RDP, RFB (VNC)	Cliente de ambiente de trabalho remoto GNOME
vinagre	V:1, I:21	4249	RDP, RFB (VNC), SPICE, SSH	Vinagre: cliente de área de trabalho remota GNOME
remmina	V:16, I:59	983	RDP, RFB (VNC), SPICE, SSH, ...	Remmina: cliente de ambiente de trabalho remoto GTK
krdc	V:2, I:15	4144	RDP, RFB (VNC)	KRDC: cliente de área de trabalho remota do KDE
virt-viewer	V:4, I:37	1278	RFB (VNC), SPICE	Cliente de visualização GUI do Gestor de Máquina Virtual do SO convidado

Tabela 7.7: Lista de servidores de acesso remoto notáveis

7.9 Ligação ao servidor X

Existem várias formas de ligar uma aplicação num anfitrião remoto ao servidor X, incluindo o `xwayland` no anfitrião local.

pacote	popcon	tamanho	comando	descrição
openssh-server	V:774, I:820	3615	`sshd` com a opção `X11-forwarding`	SSH servidor (seguro)
openssh-client	V:682, I:997	3611	`ssh -X`	SSH cliente (seguro)
xauth	V:192, I:974	81	`xauth`	Utilitário de ficheiro de autoridade X
x11-xserver-utils	V:373, I:533	559	`xhost`	controlo de acesso do servidor para X

Tabela 7.8: Lista de métodos de ligação ao servidor X

7.9.1 Ligação local do servidor X

O acesso ao servidor X local por parte das aplicações locais que utilizam o protocolo X core pode ser ligado localmente através de um socket de domínio UNIX local. Isto pode ser autorizado pelo ficheiro de autoridade que contém o

cookie de acesso. A localização do ficheiro de autoridade é identificada pela variável de ambiente "`$XAUTHORITY`" e o ecrã X é identificado pela variável de ambiente "`$DISPLAY`". Uma vez que estas são normalmente definidas automaticamente, não é necessária nenhuma ação especial, e.g. "`gitk`" como o seguinte.

```
username $ gitk
```

Nota
Para `xwayland`, `XAUTHORITY` tem um valor como "`/run/user/1000/.mutter-Xwaylandauth.YVSU30`".

7.9.2 Ligação remota ao servidor X

O acesso ao ecrã do servidor X local a partir das aplicações remotas que utilizam o protocolo central X é suportado através da utilização da funcionalidade de reencaminhamento X11.

- Abrir um `gnome-terminal` na máquina local.
- Execute ssh(1) com a opção `-X` para estabelecer uma ligação com o sítio remoto, como se segue.

```
localname @ localhost $ ssh -q -X loginname@remotehost.domain
Password:
```

- Execute um comando de aplicação X, ex. "`gitk`", no site remoto com o seguinte.

```
loginname @ remotehost $ gitk
```

Este método pode mostrar o resultado de um cliente X remoto como se ele estivesse ligado localmente através de um socket de domínio UNIX local.

Veja Secção 6.3 para SSH/SSHD.

Atenção
Uma ligação TCP/IP remota ao servidor X está desativada por predefinição no sistema Debian por razões de segurança. Não a ative definindo simplesmente "`xhost +`" nem ativando ligação XDMCP, se o puder evitar.

7.9.3 Ligação chroot do servidor X

O acesso ao servidor X por parte das aplicações que utilizam o protocolo central X e que são executadas no mesmo anfitrião, mas num ambiente como o chroot, onde o ficheiro de autoridade não está acessível, pode ser autorizado de forma segura com o `xhost`, utilizando o acesso baseado no utilizador, por exemplo, "`gitk`" como o seguinte.

```
username $ xhost + si:localuser:root ; sudo chroot /path/to
# cd /src
# gitk
# exit
username $ xhost -
```

7.10 Área de transferência (Clipboard)

Para recortar texto para a área de transferência, ver Secção 1.4.4.

Para recortar gráficos para a área de transferência, consulte Secção 11.6.

Alguns comandos CLI também podem manipular a área de transferência de caracteres (PRIMARY e CLIPBOARD).

pacote	popcon	tamanho do pacote	alvo	descrição
xsel	V:8, I:42	55	X	interface de linha de comandos para seleções X (área de transferência)
xclip	V:18, I:78	62	X	interface de linha de comandos para seleções X (área de transferência)
wl-clipboard	V:10, I:27	174	Wayland	wl-copy wl-paste: interface de linha de comando para a área de transferência do Wayland
gpm	V:8.3, I:9.0	524	Consola do Linux	um daemon que captura eventos do rato na consola Linux

Tabela 7.9: Lista de programas relacionados com a manipulação da área de transferência de caracteres

Capítulo 8

I18N e L10N

O Multilingualization (M17N) ou Suporte de Linguagem Nativa para um software de aplicação é feito em 2 passos.

- Internationalization (I18N): Para fazer com que o software lide potencialmente com múltiplos locales.
- Localization (L10N): Tornar o software útil num locale específico.

Dica
Existem 17, 18 ou 10 letras entre ”m” e ”n”, ”i” e ”n”, ou ”l” e ”n” na multi-linguagem, internacionalização e localização, que correspondem a M17N, I18N e L10N. Consulte Internacionalização e localização para obter detalhes.

8.1 O locale

O comportamento dos programas que suportam a internacionalização é configurado pela variável de ambiente ”`$LANG`” para suportar a localização. O suporte de funcionalidades dependentes da configuração regional pela biblioteca `libc` requer a instalação dos pacotes `locales` ou `locales-all`. O pacote `locales` necessita de ser inicializado corretamente.

Se nem o pacote `locales` nem o pacote `locales-all` estiverem instalados, o suporte a funcionalidades regionais perde-se e o sistema utiliza mensagens em inglês dos EUA e trata os dados como **ASCII**. Este comportamento é o mesmo que ”`$LANG`” é definido por ”`LANG=`”, ”`LANG=C`”, ou ”`LANG=POSIX`”.

O software moderno como o GNOME e o KDE são multilingue. Eles são internacionalizados ao fazê-los lidar com dados UTF-8 e localizados ao disponibilizar as suas mensagens traduzidas através da infraestrutura gettext(1). As mensagens traduzidas podem ser disponibilizadas em pacotes de configuração local independentes.

O atual sistema GUI de área de trabalho Debian normalmente define a configuração regional sob o ambiente GUI como ”`LANG=xx_YY.UTF-8`”. Aqui, ”`xx`” são códigos de idioma ISO 639 e ”`YY`” são códigos de país ISO 3166. Estes valores são definidos pela caixa de diálogo GUI de configuração do ambiente de trabalho e alteram o comportamento do programa. Veja Secção 1.5.2

8.1.1 Fundamentos para o locale UTF-8

A representação mais simples dos dados de texto é o **ASCII**, que é suficiente para o inglês e utiliza menos de 127 caracteres (representáveis com 7 bits).

Mesmo o Inglês simples pode conter caracteres não-ASCII, ex. as marcas de citação esquerda e direita ligeiramente curvas não estão disponíveis em ASCII.

```
b''“b''double quoted textb''”b'' is not "double quoted ASCII"
b''‘b''single quoted textb''’b'' is not 'single quoted ASCII'
```

A fim de suportar mais caracteres, foram utilizados muitos conjuntos de caracteres e sistemas de codificação para suportar muitas línguas (ver Tabela 11.2).

O conjunto de caracteres Unicode pode representar praticamente todos os caracteres conhecidos pelo ser humano com uma gama de pontos de código de 21 bits (ou seja, de 0 a 10FFFF em notação hexadecimal).

O sistema de codificação de texto UTF-8 adapta os pontos de código Unicode a um fluxo de dados de 8 bits sensato, compatível sobretudo com o sistema de processamento de dados ASCII. Este facto torna o **UTF-8** a escolha mais moderna e preferida. **UTF** significa Unicode Transformation Format (Formato de Transformação Unicode). Quando os dados de texto simples ASCII são convertidos para UTF-8, têm exatamente o mesmo conteúdo e tamanho que o ASCII original. Por isso, não perde nada ao implementar a configuração regional UTF-8.

Sob a configurações regionais UTF-8 com programa de aplicação compatível, pode apresentar e editar dados de texto em qualquer língua estrangeira desde que as fontes e os métodos de introdução necessários estejam instalados e ativados. Por exemplo, sob a configuração regional ”`LANG=fr_FR.UTF-8`”, o gedit(1) (editor de texto para o ambiente de trabalho GNOME) pode mostrar e editar dados de texto em caracteres chineses enquanto apresenta menus em francês.

Dica
Tanto a nova configuração regional padrão ”`en_US.UTF-8`” como a antiga configuração regional padrão ”`C`”/”`POSIX`” utilizam a mensagem padrão do Inglês dos EUA, têm diferenças subtis na ordem de ordenação, etc. Se quiser lidar não só com caracteres ASCII mas também com todos os caracteres codificados em UTF-8 graciosamente enquanto mantém o antigo comportamento local ”`C`”, use em Debian a configuração regional não-padrão ”`C.UTF-8`”.

Nota
Alguns programas consomem mais memória após suportarem I18N. Isto é porque estão codificados para usar UTF-32(UCS4) internamente para suportar Unicode para optimização de velocidade e consomem 4 bytes por cada caractere ASCII independentemente do locale selcionado. Mais uma vez, não perde nada ao implantar o locale UTF-8.

8.1.2 A reconfiguração do locale

Para que o sistema aceda a uma determinada configuração regional, os dados dessa localização devem ser compilados a partir da base de dados das configurações regionais.

O pacote `locales` **não** vem com dados de localidade pré-compilados. É necessário configurá-lo como:

```
# dpkg-reconfigure locales
```

Este processo involve 2 passos.

1. Selecione todas as configurações regionais necessárias para serem compiladas no formato binário. (Certifique-se de que inclui pelo menos uma configuração UTF-8)
2. Definir a configuração regional predefinida de todo o sistema em ”`/etc/default/locale`” para usar com o PAM (veja Secção 4.5).

O valor da configuração regional por defeito do sistema definido em ”`/etc/default/locale`” pode ser substituído pela configuração GUI para aplicações GUI.

Nota
O sistema de codificação tradicional atual pode ser identificado por "`/usr/share/i18n/SUPPORTED`". Assim, "`LANG=pt`" é "`LANG=pt.ISO-8859-1`".

O pacote `locales-all` vem com dados de regionais pré-compilados para todos os dados de região. Uma vez que não cria o ficheiro "`/etc/default/locale`", poderá ter de instalar também o pacote `locales`.

Dica
O pacote `locales` de algumas distribuições derivadas de Debian vêm com dados da configuração de região pré-compilados para todos os dados regionais. Você precisa de instalar ambos os pacotes `locales` e `locales-all` em Debian para emular esse ambiente de sistema.

8.1.3 Codificação de nomes de ficheiros

Para troca de dados entre plataformas (veja Secção 10.1.7), pode precisar de montar algum sistema de ficheiros com codificações particulares. por exemplo, o mount(8) para sistema de ficheiros vfat assume CP437 se usado sem opção. Precisa de fornecer uma opção explícita de montagem para usar UTF-8 ou CP932 para os nomes dos ficheiros.

Nota
Quando se monta automaticamente uma Memória flash USB sob ambientes de trabalho modernos como o GNOME, pode fornecer tal opção de montagem ao clicar com o botão direito no ícone no ambiente de trabalho, clique no separador "Drive", clique para expandir "Definições" e insira "utf8" nas "Opções de montagem:". Não próxima vez que esta memória flash USB for montada, está activa a montagem com UTF-8.

Nota
Se está a atualizar o sistema ou a mover os discos de um sistema antigo não-UTF-8, os nomes de ficheiros com caracteres não-ASCII podem ser codificados com as codificações históricas e obsoletas como a ISO-8859-1 ou eucJP. por favor procure a ajuda de ferramentas de conversão de texto para convertê-los para UTF-8. Veja Secção 11.1.

O Samba usa Unicode para os clientes mais recentes (Windows NT, 200x, XP) mas usa CP850 para os clientes mais antigos (DOS e Windows 9x/Me) por predefinição. Esta predefinição para os clientes mais antigos pode ser alterada a usar "`dos charset`" no ficheiro "`/etc/samba/smb.conf`" por exemplo para CP932 para Japonês.

8.1.4 Mensagens localizadas e documentação traduzida

Existem traduções para muitas das mensagens de texto e documentos que são mostrados no sistema Debian, tais como as mensagens de erro, as saídas normais dos programas, os menus e os manuais. A cadeia de ferramentas de comandos gettext(1) do GNU é usada como a ferramenta backend para a maioria das atividades de tradução.

Sob "Tarefas" → "Localização" o aptitude(8) disponibiliza uma lista extensa de pacotes binários úteis que adicionam mensagens localizadas às aplicações e fornecem documentação traduzida.

Por exemplo, pode obter as mensagens localizadas para os manuais ao instalar o pacote `manpages-`*`LANG`*. Para ler o manual em linguagem Italiana para o *nome_do_programa* a partir de "`/usr/share/man/it/`", execute o seguinte.

```
LANG=it_IT.UTF-8 man programname
```

O GNU gettext pode acomodar uma lista prioritária de idiomas de tradução com a variável de ambiente `$LANGUAGE`. Por exemplo:

```
$ export LANGUAGE="pt:pt_BR:es:it:fr"
```

Para mais informações, consulte `info gettext` e leia a secção ”A variável LANGUAGE”.

8.1.5 Efeitos do locale

A ordem de ordenação dos caracteres com sort(1) e ls(1) é afetada pelas configurações regionais. Exportando `LANG=en_US.UTF-8` ordena pelo dicionário `A->a->B->b...->Z->z`, enquanto a exportação de `LANG=C.UTF-8` ordena na ordem binária `A->B->...->Z->a->b...` do dicionário.

O formato de data de ls(1) é afetado pela configuração regional (ver Secção 9.3.4).

O formato de data de date(1) é afetado pela configuração regional. Por exemplo:

```
 $ unset LC_ALL
 $ LANG=en_US.UTF-8 date
Thu Dec 24 08:30:00 PM JST 2023
 $ LANG=en_GB.UTF-8 date
Thu 24 Dec 20:30:10 JST 2023
 $ LANG=es_ES.UTF-8 date
jue 24 dic 2023 20:30:20 JST
 $ LC_TIME=en_DK.UTF-8 date
2023-12-24T20:30:30 JST
```

As pontuações numéricas são diferentes consoante as configurações locais. Por exemplo, na configuração regional Inglês, mil virgula um é mostrado como ”`1,000.1`” enquanto na configuração regional Alemão é mostrado como `1.000,1`”. Pode ver esta diferença num programa de folha de cálculo.

Cada caraterística pormenorizada da variável de ambiente ”`$LANG`” pode ser substituída pela definição das variáveis ”`$LC_*`”. Estas variáveis de ambiente podem ser substituídas novamente pela definição da variável ”`$LC_ALL`”. Veja a página de manual do locale(7) para mais detalhes. A não ser que tenha uma razão forte para criar configurações complicadas, por favor mantenha-se afastado delas e use apenas a variável ”`$LANG`” definida para uma das configurações regionais UTF-8.

8.2 A entrada do teclado

O sistema do teclado pode ser configurado em diferentes disposições do sistema.

- Kernel Linux: keyboard(5)
- Servidor X: setxkbmap(1), xkeyboard-config(5), variável de ambiente `XMODIFIERS`
- GUI de ambiente de trabalho: Estrutura de Método de Entrada: `ibus`, `fcitx5`
- Aplicação: variáveis de ambiente para definir a sua fonte de entrada: `GTK_IM_MODULE`, `QT_IM_MODULE`, `QT_IM_MODULE`
  ...

A estrutura de método de entrada (IM) consiste de:

- Motor de método de entrada (IME): Método de entrada atual
- Configuração: Lida com a configuração para IBus e outros serviços tais como plugins de IME
- Painel: Interface de utilizador como uma barra de linguagem e tabela de seleção de candidado

Entrada multi-linguagem para a aplicação é processada mais ou menos como:

```
Keyboard        UI panel    Configuration          Application
|                  ^ |           |                       ^  ^
v                  | v           v                       |  |
Linux kernel -> Input method engine (IME) -+-> Gtk, Qt ---+  |
                   | ^                     |                 |
                   | |                     +-> X11, Wayland -+
                   v |
                IME plugin (ibus-mozc, ...)
```

8.2.1 A entrada de teclado para a consola Linux e o X Window

O sistema Debian pode ser configurado para funcionar com muitas disposições internacionais de teclado ao usar o pacote `keyboard-configuration` .

```
# dpkg-reconfigure keyboard-configuration
```

Para a consola Linux e o sistema X Window, isto atualiza os parâmetros de configuração em "`/etc/default/keyboard`". Muitos caracteres não-ASCII incluindo caracteres acentuados usados por muitas línguas Europeias podem ser disponibilizados com a tecla dead, a tecla AltGr, e a tecla compose.

Nota
Se o ibus estiver ativo, a sua configuração clássica do teclado X através do `setxkbmap` pode ser sobreposta pelo `ibus` mesmo sob o ambiente de trabalho clássico baseado no X. Pode desativar o `ibus` instalado utilizando o `im-config` para definir o método de entrada para "None". Para mais, veja Debian Wiki sobre teclado.

8.2.2 A entrada de teclado para o Wayland

Ao contrário do protocolo X Window, o protocolo núcleo do Wayland nem sequer suporta a entrada de caracteres acentuados. Compositores Populares do Wayland, como o Mutter para GNOME ou KWin para KDE, implementam protocolos de extensão como o `text-input-unstable-v3` para a entrada de texto (veja "Protocolos Wayland atuais e o seu estado de suporte").

O protocolo `text-input-unstable-v3` funciona bem com métodos de entrada para Wayland (veja "Análise retrospetiva de projeto de método de entrada de Wayland").

- A maioria das aplicações GUI são compiladas com bibliotecas GUI como a GTK ou Qt que suportam este `text-input-un`
- Motores de Método de Entrada (IME) Populares, com o IBus ou Fcitx (versão 5), podem trabalhar com este `text-input-unstable-v3`.
- As IMEs suportam entrada de texto para muitas linguagens com plugins.
- Os IMEs recentes integram funcionalidades de "X Keyboard Extensão (XKB)" como a `setxkbmap` anteriormente fornecida pelo X Window para suportar entrada de texto de caracteres acentuados para linguagens Europeias para o Wayland.

8.2.3 O suporte a método de entrada com IBus

Para GNOME, o "`ibus`" é o IME predefinido o qual é instalado automaticamente via a sua dependência de pacote.

- A maioria das funcionalidades de entrada de teclado de multi-linguagem podem ser configuradas em "Definições do GNOME" ou "Ajustes do GNOME".

- Algumas funcionalidades de entrada de teclado multi-linguagem podem precisar de ser configuradas a partir do comando ibus-setup(1).
- A entrada de teclado emoji está disponível ao teclar ”SUPER-.” (Em simultâneo teclar Windows e teclas de ponto) seguido de uma palavra chave para cada emoji e teclas de ESPAÇO.

A lista do IBus e dos seus pacotes plugin é a seguinte.

pacote	popcon	tamanho	locale suportado
ibus	V:212, I:253	1884	estrutura de método de entrada que usa dbus
ibus-mozc	V:2.1, I:3.8	1054	Japonês
ibus-anthy	V:0.5, I:1.1	8740	Japonês
ibus-skk	V:0.03, I:0.14	242	Japonês
ibus-kkc	V:0.04, I:0.19	206	Japonês
ibus-libpinyin	V:1.1, I:5.1	123	Chinês (para zh_CN)
ibus-chewing	V:0.16, I:0.84	298	Chinês (para zh_TW)
ibus-libzhuyin	I:0.10	41021	Chinês (para zh_TW)
ibus-rime	V:0.29, I:0.51	76	Chinês (para zh_CN/zh_TW)
ibus-cangjie	V:0.02, I:0.10	235	Chinês (para zh_HK)
ibus-hangul	V:0.3, I:1.9	261	Coreano
ibus-libthai	V:0.00, I:0.05	84	Thai
ibus-table-thai	I:0.04	59	Thai
ibus-unikey	V:0.22, I:0.40	286	Vietnamita
keyman	V:0.01, I:0.10	507	Multi-linguagem: plugin Keyman para mais de 2000 idiomas
ibus-table	V:0.04, I:0.92	2364	tabela de plugins para IBus
ibus-m17n	V:0.1, I:1.8	373	Multilingue: Indiano, Árabe e outros

Tabela 8.1: Lista do IBus e dos seus pacotes plugin

8.2.4 O suporte de método de entrada com Fcitx

A estrutura de método de entrada Fcitx (versão 5) é popular com os utilizadores Chineses e compatível com ”`ibus`”.

A lista do ”`fcitx5`” e dos seus pacotes plugin é a seguinte.

8.2.5 Um exemplo para Japonês

Acho muito útil o método de entrada de Japonês começado sob ambiente Inglês ”`en_US.UTF-8`”). Aqui está como fiz isto com o IBus:

1. Instale o pacote de ferramenta de entrada em japonês `ibus-mozc` (ou `ibus-anthy`).
2. • Genérico: Execute ibus-setup(1) → selecione ”Método de Entrada” → clique ”`Adicionar`” → ”Japonês” → ”Mozc (ou Anthy)” → clique ”Adicionar”

pacote	popcon	tamanh	docale suportado
fcitx5	V:8, I:13	712	estrutura de método de entrada compatível com "ibus"
fcitx5-mozc	V:1.1, I:1.8	1328	Japonês
fcitx5-anthy	V:0.07, I:0.25	853	Japonês
fcitx5-skk	V:0.05, I:0.15	370	Japonês
fcitx5-kkc	V:0.01, I:0.06	418	Japonês
fcitx5-chinese-addons	I:9.2	17	Chinês (meta-pacote para zh_*)
fcitx5-pinyin	V:4.0, I:9.7	1000	Chinês (para zh_CN)
fcitx5-chewing	V:0.29, I:0.98	228	Chinês (para zh_TW)
fcitx5-zhuyin	I:0.08	108535	Chinês (para zh_TW)
fcitx5-rime	V:0.46, I:0.93	393	Chinês (para zh_CN/zh_TW)
fcitx5-table-cangjie-large	I:0.13	1292	Chinês (para zh_HK)
fcitx5-hangul	V:0.12, I:0.27	239	Coreano
fcitx5-libthai	V:0.00, I:0.04	119	Thai
fcitx5-table-thai	I:0.09	34	Thai
fcitx5-unikey	V:0.13, I:0.23	624	Vietnamita
fcitx5-m17n	V:0.16, I:0.61	171	Multilingue: Indiano, Árabe e outros
fcitx5-table	V:0.3, I:9.6	428	tabela de plugins para fcitx5
fcitx5-keyman	V:0.04, I:0.05	241	Multi-linguagem: plugin Keyman para mais de 2000 idiomas

Tabela 8.2: Lista do Fcitx5 e dos seus pacotes plugin

    - GNOME: Selecione ”Definições” → ”Teclado” → ”Fontes de Entrada” → clique ”+” em ”Fontes de Entrada” → ”Japonês” → ”Mozc (ou Anthy)” → clique ”Adicionar”
3. Pode escolher tantas fontes de entrada quantas desejar.
4. Voltar a fazer login na conta do utilizador.
5. Configure cada fonte de entrada clicando com o botão direito do rato no ícone da barra de ferramentas GUI.
6. Alterne entre as fontes de entrada instaladas com SUPER-ESPAÇO. (SUPER é normalmente a Tecla Windows.)

Dica
Se pretender aceder ao ambiente de teclado só com o alfabeto com o teclado físico japonês no qual shift-2 tem gravado " (aspas duplas), selecione ”Japonês” no procedimento acima. Pode introduzir a língua japonesa utilizando ”mozc Japonês (ou anthy)” com o teclado físico ”US” no qual shift-2 tem gravado @ (arroba).

- Para o Wayland:
  - O pacote `im-config` não faz nada e pode ser removido em segurança.
  - Provavelmente não vai precisar de definir variáveis de ambiente excepto para compatibilidade com coisas antigas, etc.
  - Se precisar de definir variáveis de ambiente, crie um ficheiro como ”`~/.config/environment.d/50-input-metho`” para as definir.
- Para X Window:
  - Instalar o pacote `im-config`.
  - A entrada do menu GUI para im-config(8) é ”Input method” (Método de entrada).
  - Em alternativa, execute ”`im-config`” a partir da shell do utilizador.
  - O im-config(8) comporta-se de modo diferente se o comando for executado pelo root ou não.
  - im-config(8) activa o melhor método de entrada do sistema e é predefinido sem qualquer acção do utilizador.

8.3 O ecrã de resultados

A consola do Linux apenas pode mostrar caracteres limitados. (Precisa usar programas de terminal especiais como o jfbterm(1) para mostrar linguagens não-Europeias numa consola não GUI.)

O ambiente GUI (Capítulo 7) pode apresentar quaisquer caracteres em UTF-8, desde que as fontes necessárias estejam instaladas e ativadas. (A codificação dos dados originais do tipo de letra é tratada e transparente para o utilizador.)

8.3.1 A configuração do terminal

O sistema Debian pode ser configurado para funcionar com muitas disposições internacionais de consola ao usar o pacote `console-setup`.

```
# dpkg-reconfigure console-setup
```

Para a consola Linux e o sistema X Window, isto atualiza os parâmetros de configuração em ”`/etc/default/console-se`” para mostrar muitos caracteres não-ASCII incluindo caracteres acentuados usados por muitas línguas Europeias que podem ser disponibilizados.

Existem vários componentes para configurar a consola de caracteres e as funcionalidades do sistema ncurses(3).

- O ficheiro "`/etc/terminfo/*/*`" (terminfo(5))
- A variável de ambiente "`$TERM`" (term(7))
- setterm(1), stty(1), tic(1) e toe(1)

Se a entrada `terminfo` para o `xterm` não funcionar com um `xterm` não Debian, mude o seu tipo e terminal, "`$TERM`", de "`xterm`" para uma das versões de funcionalidades limitadas como o "`xterm-r6`" quando iniciar sessão num sistema Debian remotamente. Veja "`/usr/share/doc/libncurses5/FAQ`" para mais. O "`dumb`" é o denominador comum mais baixo para "`$TERM`".

8.3.2 Caracteres Asiáticos de Leste de Altura Ambígua

Em localizações ao leste da Asia, o desenhar de caixa, os caracteres Gregos e Cirílicos podem aparecer mais largos que o desejado e causarem o desalinhamento do resultado do terminal (veja Standard Unicode Annex #11, 4.2 Caracteres Ambíguos).

Pode contornar este problema:

- `gnome-terminal`: Preferências → Perfis → *nome do perfil* → Compatibilidade → Caracteres de Largura Ambígua → Estreito
- `ncurses`: Define ambiente `export NCURSES_NO_UTF8_ACS=0`.

Capítulo 9

Dicas do sistema

Aqui, descrevo dicas básicas para configurar e gerir sistemas, a maioria a partir da consola.

9.1 As dicas da consola

Existem alguns programas utilitários para ajudar as atividades da consola.

pacote	popcon	tamanho	descrição
mc	V:40, I:176	1590	Veja Secção 1.3
bsdutils	V:421, I:999	338	script(1) comando para fazer um registo da sessão de terminal
screen	V:50, I:191	1007	terminal multiplexador com emulação de terminal VT100/ANSI
tmux	V:106, I:167	1423	multiplexador alternativo de terminal (Use ”Control-B” em vez disto)
ripgrep	V:17, I:48	5284	pesquisa rápida e recursiva de cadeias de caracteres na árvore de código-fonte com filtragem automática
fzf	V:10, I:36	5677	localizador de texto difuso
fzy	V:0.07, I:0.34	59	localizador de texto difuso
rlwrap	V:1, I:11	328	funcionalidade de revestimento e histórico da linha de comando
ledit	V:0.5, I:7.3	375	funcionalidade de revestimento e histórico da linha de comando
rlfe	V:0.02, I:0.48	45	funcionalidade de revestimento e histórico da linha de comando

Tabela 9.1: Lista de programas de apoio às atividades da consola

9.1.1 Gravar as atividades da shell de modo limpo

O uso simples de script(1) (veja Secção 1.4.9) para gravar a atividade da shell produz um ficheiro com caracteres de controle. Isto pode ser evitado ao usar o col(1) como o seguinte.

```
$ script
Script started, file is typescript
```

Faça o que tem a fazer ... e carregue em `Ctrl-D` para terminar o `script`.

```
$ col -bx < typescript > cleanedfile
$ vim cleanedfile
```

Existem métodos alternativos para registar as atividades da shell:

- Use `tee` (utilizável durante o processo de arranque no initramfs):

```
$ sh -i 2>&1 | tee typescript
```

- Use o `gnome-terminal` com a memória intermédia de linha estendida para conseguir rolar para trás.
- Utilize o `ecrã` com "`^A H`" (ver Secção 9.1.2) para efetuar a gravação da consola.
- Utilize o `vim` com "`:terminal`" para entrar no modo de terminal. Use "`Ctrl-W N`" para sair do modo terminal para o modo normal. Utilize "`:w typescript`" para escrever a memória intermédia num ficheiro.
- Use o `emacs` com "`M-x shell`", "`M-x eshell`", ou "`M-x term`" para entrar na consola de gravação. Use"`C-x C-w`" para escrever a memória intermédia para um ficheiro.

9.1.2 O programa screen

O screen(1) não apenas permite que uma janela terminal funcione com múltiplos processos, mas também permite que os **processos de shell remota sobrevivam a ligações interrompidas**. Aqui está um cenário típico de utilização do screen(1).

1. Faz login numa máquina remota.
2. Arranca o `screen` numa consola única.
3. Executa múltiplos programas na janela criada do `screen` com `^A c` ("Ctrl-A" seguido de "c").
4. Muda entre as múltiplas janelas do `screen` com `^A n` ("Ctrl-A" seguido de "n").
5. Subitamente precisa de abandonar o seu terminal, mas não quer perder o seu trabalho ativo a manter a ligação.
6. Pode **separar** a sessão do `screen` por quaisquer métodos.
   - Desligar a sua ligação de rede à bruta
   - Escrever `^A d` ("Ctrl-A" seguido de "d") e manualmente terminar a sessão da ligação remota
   - Escrever `^A DD` ("Ctrl-A" seguido de "DD") para separar o `screen` e terminar a sua sessão
7. Faz login de novo à mesma máquina remota (mesmo a partir de um terminal diferente).
8. Inicia o `screen` como "`screen -r`".
9. O `screen` magicamente **reagrupa** todas as janelas `screen` anteriores com todos os programas a funcionar activamente.

Dica
Pode poupar despesas de ligação com o `screen` em ligações de rede medidas 'a metro' como as dial-up, porque pode deixar um processo ativo enquanto desligado e depois lhe re-ligar-se mais tarde quando ligar de novo.

Numa sessão do `screen`, todas as entradas do teclado são enviadas à sua janela atual excepto as teclas de comandos. Todas as teclas de comando do `screen` são inseridas ao escrever `^A` ("Control-A") mais uma única tecla [mais quaisquer parâmetros]. Aqui estão alguns importantes para fazer lembrar.

Veja screen(1) para detalhes.

Veja tmux(1) para as funcionalidades do comando alternativo.

tecla de atalho	significado
^A ?	mostra um écran de ajuda (mostra as teclas de atalho)
^A c	cria uma nova janela e muda para lá
^A n	vai à janela seguinte
^A p	vai à janela anterior
^A 0	vaia à janela número 0
^A 1	vai à janela número 1
^A w	mostra uma lista de janelas
^A a	envia Ctrl-A à janela atual como entrada do teclado
^A h	escreve uma cópia física da janela atual para um ficheiro
^A H	inicia/termina o registo da janela atual para um ficheiro
^A ^X	bloqueia o terminal (protegido por palavra-passe)
^A d	separa a sessão do ecrã do terminal
^A DD	separa a sessão do ecrã e termina a sessão

Tabela 9.2: Lista de ligações de teclas para o screen

9.1.3 Navegando nos diretórios

Em Secção 1.4.2, são descritas 2 dicas para permitir uma navegação rápida pelos diretórios: `$CDPATH` e `mc`.

Se utilizar o programa de filtragem de texto fuzzy, pode fazê-lo sem escrever o caminho exato. Para o `fzf`, inclua o seguinte em `~/.bashrc`.

```
FZF_KEYBINDINGS_PATH=/usr/share/doc/fzf/examples/key-bindings.bash
if [ -f $FZF_KEYBINDINGS_PATH ]; then
  . $FZF_KEYBINDINGS_PATH
fi
```

Por exemplo:

- Pode saltar para um sub-diretório muito profundo com esforço mínimo. Primeiro, escreva "`cd **`" e prima `Tab`. Em seguida, ser-lhe-ão solicitados os caminhos candidatos. Escrever uma sequência de caminhos parciais, ex, `s/d/b foo`, irá reduzir os caminhos candidatos. Você seleciona o caminho a ser utilizado por `cd` com o cursor e as teclas de retorno.
- É possível selecionar um comando do histórico de comandos de forma mais eficiente e com o mínimo de esforço. Prima `Ctrl-R` na linha de comandos. Em seguida, ser-lhe-ão apresentados os comandos candidatos. Escrever cadeias de comandos parciais, e.g., `vim d`, irá reduzir os candidatos. Seleciona o comando a ser utilizado com as teclas de cursor e de retorno.

9.1.4 Revestimento da linha de leitura

Alguns comandos, como o `/usr/bin/dash`, que não tem capacidade de edição do histórico da linha de comandos, podem adicionar essa funcionalidade de forma transparente ao serem executados sob o `rlwrap` ou seus equivalentes.

```
$ rlwrap dash -i
```

Isto fornece uma plataforma conveniente para testar pontos subtis para `dash` com um ambiente amigável semelhante ao `bash`.

9.1.5 Verificação da árvore de código-fonte

O comando rg(1) do pacote `ripgrep` oferece uma alternativa mais rápida ao comando grep(1) para varrer a árvore de código-fonte em situações típicas. Ele tira vantagem das modernas CPUs multi-core e automaticamente aplica filtros razoáveis para pular alguns arquivos.

9.2 Personalizar o vim

Depois de aprender as noções básicas do vim(1) através do Secção 1.4.8, por favor leia o livro de Bram Moolenaar "Seven habits of effective text editing (2000)" para compreender como o `vim` deve ser utilizado.

9.2.1 Personalizando o vim com recursos internos

O comportamento do `vim` pode ser alterado significativamente ativando as suas caraterísticas internas através dos comandos do modo Ex tais como "`set ...`" para definir as opções do vim.

Estes comandos em modo Ex podem ser incluídos no ficheiro vimrc do utilizador, no tradicional "`~/.vimrc`" ou no git-friendly "`~/.vim/vimrc`". Aqui está um exemplo muito simples 1:

```
""" Generic baseline Vim and Neovim configuration (~/.vimrc)
"""   - For NeoVim, use "nvim -u ~/.vimrc [filename]"
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
let mapleader = ' '             " :h mapleader
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
set nocompatible                " :h 'cp -- sensible (n)vim mode
syntax on                       " :h :syn-on
filetype plugin indent on       " :h :filetype-overview
set encoding=utf-8              " :h 'enc (default: latin1) -- sensible encoding
""" current vim option value can be verified by :set encoding?
set backspace=indent,eol,start  " :h 'bs (default: nobs) -- sensible BS
set statusline=%<%f%m%r%h%w%=%y[U+%04B]%2l/%2L=%P,%2c%V
set listchars=eol:¶,tab:b''▫b''\ ,extends:b''▫b'',precedes:b''▫b'',nbsp:b''▫b''
set viminfo=!,'100,<5000,s100,h " :h 'vi -- bigger copy buffer etc.
""" Pick "colorscheme" from blue darkblue default delek desert elflord evening
""" habamax industry koehler lunaperche morning murphy pablo peachpuff quiet ron
""" shine slate torte zellner
colorscheme industry
""" don't pick "colorscheme" as "default" which may kill SpellUnderline settings
set scrolloff=5                 " :h 'scr -- show 5 lines around cursor
set laststatus=2                " :h 'ls (default 1)  k
""" boolean options can be unset by prefixing "no"
set ignorecase                  " :h 'ic
set smartcase                   " :h 'scs
set autoindent                  " :h 'ai
set smartindent                 " :h 'si
set nowrap                      " :h 'wrap
"set list                        " :h 'list (default nolist)
set noerrorbells                " :h 'eb
set novisualbell                " :h 'vb
set t_vb=                       " :h 't_vb -- termcap visual bell
set spell                       " :h 'spell
set spelllang=en_us,cjk         " :h 'spl -- english spell, ignore CJK
set clipboard=unnamedplus       " :h 'cb -- cut/copy/paste with other app
set hidden                      " :h 'hid
set autowrite                   " :h 'aw
set timeoutlen=300              " :h 'tm
```

O mapa de teclas do `vim` pode ser alterado no ficheiro vimrc do utilizador. Por exemplo:



Cuidado
Não tente alterar os atalhos de teclas predefinidos sem ter boas razões para o fazer.

1Exemplos de personalização mais elaborados: "Vim Galore", "sensible.vim", ...

```
""" Popular mappings (imitating LazyVim etc.)
""" Window moves without using CTRL-W which is dangerous in INSERT mode
nnoremap <C-H> <C-W>h
nnoremap <C-J> <C-W>j
nnoremap <C-K> <C-W>k
silent! nnoremap <C-L> <C-W>l
""" Window resize
nnoremap <C-LEFT> <CMD>vertical resize -2<CR>
nnoremap <C-DOWN> <CMD>resize -2<CR>
nnoremap <C-UP> <CMD>resize +2<CR>
nnoremap <C-RIGHT> <CMD>vertical resize +2<CR>
""" Clear hlsearch with <ESC> (<C-L> is mapped as above)
nnoremap <ESC> <CMD>noh<CR><ESC>
inoremap <ESC> <CMD>noh<CR><ESC>
""" center after jump next
nnoremap n nzz
nnoremap N Nzz
""" fast "jk" to get out of INSERT mode (<ESC>)
inoremap  jk <CMD>noh<CR><ESC>
""" fast "<ESC><ESC>" to get out of TERM mode (CTRL-\ CTRL-N)
tnoremap <ESC><ESC> <C-\><C-N>
""" fast "jk" to get out of TERM mode (CTRL-\ CTRL-N)
tnoremap jk <C-\><C-N>
""" previous/next trouble/quickfix item
nnoremap [q <CMD>cprevious<CR>
nnoremap ]q <CMD>cnext<CR>
""" buffers
nnoremap <S-H> <CMD>bprevious<CR>
nnoremap <S-L> <CMD>bnext<CR>
nnoremap [b <CMD>bprevious<CR>
nnoremap ]b <CMD>bnext<CR>
""" Add undo break-points
inoremap  , ,<C-G>u
inoremap  . .<C-G>u
inoremap  ; ;<C-G>u
""" save file
inoremap <C-S> <CMD>w<CR><ESC>
xnoremap <C-S> <CMD>w<CR><ESC>
nnoremap <C-S> <CMD>w<CR><ESC>
snoremap <C-S> <CMD>w<CR><ESC>
""" better indenting
vnoremap < <gv
vnoremap > >gv
""" terminal (Somehow under Linux, <C-/> becomes <C-_> in Vim)
nnoremap <C-_> <CMD>terminal<CR>
"nnoremap <C-/> <CMD>terminal<CR>
""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
if ! has('nvim')
""" Toggle paste mode with <SPACE>p for Vim (no need for Nvim)
set pastetoggle=<leader>p
""" nvim default mappings for Vim.  See :h default-mappings in nvim
""" copy to EOL (no delete) like D for d
noremap Y y$
""" sets a new undo point before deleting
inoremap <C-U> <C-G>u<C-U>
inoremap <C-W> <C-G>u<C-W>
""" <C-L> is re-purposed as above
""" execute the previous macro recorded with Q
nnoremap Q @@
""" repeat last substitute and *KEEP* flags
nnoremap & :&&<CR>
```

```
""" search visual selected string for visual mode
xnoremap * y/\V<C-R>"<CR>
xnoremap # y?\V<C-R>"<CR>
endif
```

Para que as combinações de teclas acima funcionem corretamente, o programa do terminal tem de ser configurado para gerar ”ASCII DEL” para a tecla `Backspace` e ”Escape sequence” para a tecla `Delete`.

Outras configurações diversas podem ser alteradas no ficheiro vimrc do utilizador. Por exemplo:

```
""" Use faster 'rg' (ripgrep package) for :grep
if executable("rg")
  set grepprg=rg\ --vimgrep\ --smart-case
  set grepformat=%f:%l:%c:%m
endif
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
""" Retain last cursor position :h '"
augroup RetainLastCursorPosition
  autocmd!
  autocmd BufReadPost *
    \ if line("'\"") > 0 && line ("'\"") <= line("$") |
    \   exe "normal! g'\"" |
    \ endif
augroup END
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
""" Force to use underline for spell check results
augroup SpellUnderline
  autocmd!
  autocmd ColorScheme * highlight SpellBad term=Underline gui=Undercurl
  autocmd ColorScheme * highlight SpellCap term=Underline gui=Undercurl
  autocmd ColorScheme * highlight SpellLocal term=Underline gui=Undercurl
  autocmd ColorScheme * highlight SpellRare term=Underline gui=Undercurl
augroup END
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
""" highlight tailing spaces except when typing as red (set after colorscheme)
highlight TailingWhitespaces ctermbg=red guibg=red
""" \s\+     1 or more whitespace character: <Space> and <Tab>
""" \%#\@<!  Matches with zero width if the cursor position does NOT match.
match TailingWhitespaces /\s\+\%#\@<!$/
```

9.2.2 Personalizando o vim com pacotes externos

Podem ser encontrados pacotes de plug-ins externos interessantes:

- Vim - o editor de texto ubíquo -- O site oficial do Vim e dos scripts vim
- VimAwsome -- A listagem de plugins Vim
- vim-scripts -- Pacote Debian: uma coleção de scripts vim

Os pacotes de plugins no pacote vim-scripts podem ser ativados usando o ficheiro vimrc do utilizador. Por exemplo:

```
packadd! secure-modelines
packadd! winmanager
" IDE-like UI for files and buffers with <space>w
nnoremap <leader>w         :WMToggle<CR>
```

O novo sistema de pacotes nativo do Vim funciona muito bem com ”`git`” e ”`git submodule`”. Um exemplo de configuração pode ser encontrado no meu repositório git: dot-vim. Isto faz essencialmente o seguinte:

- Ao utilizar "`git`" e "`git submodule`", os pacotes externos mais recentes, tais como "*name*", são colocados em `~/.vim/pack/*/opt/`*name* e similares.
- Ao adicionar `:packadd!` *name* ao ficheiro vimrc do utilizador, estes pacotes são colocados no `runtimepath`.
- O Vim carrega estes pacotes no `runtimepath` durante a sua inicialização.
- No final da sua inicialização, as etiquetas dos documentos instalados são atualizadas com "`helptags ALL`".

Para mais, por favor inicie o `vim` com "`vim --startuptime vimstart.log`" para verificar a sequência de execução atual e o tempo gasto em cada passo.

É bastante confuso ver demasiadas formas[2] de gerir e carregar estes pacotes externos para o `vim`. Verificar a informação original é a melhor solução.

toques de tecla	**informação**
`:help package`	explicação sobre o mecanismo de pacotes do vim
`:help runtimepath`	explicação sobre o mecanismo de `runtimepath`
`:version`	estados internos incluindo candidatos para o ficheiro vimrc
`:echo $VIM`	a variável de ambiente "`$VIM`" utilizada para localizar o ficheiro vimrc
`:set runtimepath?`	lista de diretórios que serão pesquisados para todos os ficheiros de suporte de tempo de execução
`:echo $VIMRUNTIME`	a variável de ambiente "`$VIMRUNTIME`" utilizada para localizar vários ficheiros de suporte de tempo de execução fornecidos pelo sistema

Tabela 9.3: Informações sobre a inicialização do `vim`

9.3 Gravação de dados e apresentação

9.3.1 O daemon de log

Muitos programas tradicionais registam as suas atividades no formato de ficheiro de texto no diretório "`/var/log/`".

logrotate(8) é utilizado para simplificar a administração de ficheiros de registo num sistema que gera muitos ficheiros de registo.

Muitos programas novos registam as suas atividades no formato de ficheiro binário utilizando o serviço de diário systemd-journald(8) no diretório "`/var/log/journal`".

É possível registar dados no diário systemd-journald(8) a partir de um script shell utilizando o comando systemd-cat(1).

Veja Secção 3.5 e Secção 3.4.

9.3.2 Analisador de relatório (Log)

Aqui estão analisadores de relatórios notáveis ("`~Gsecurity::log-analyzer`" no aptitude(8)).

Nota
CRM114 disponibiliza uma infraestrutura de linguagem para escrever filtros **fuzzy** com a biblioteca de expressões regulares TRE. O uso popular dela é o filtro de spam de mail, mas pode ser usado como um analisador de registos.

2vim-pathogen foi popular.

pacote	popcon	tamanho	descrição
fail2ban	V:101, I:111	2275	banir IPs que causam vários erros de autenticação
logwatch	V:9, I:10	2440	analisador de log com saída bonita escrito em Perl
awstats	V:5.1, I:8.0	6934	analisador de logs de servidor web poderoso e cheio de funcionalidades
analog	V:3, I:83	3739	analisador de log do servidor web
sarg	V:0.65, I:0.71	863	gerador de relatórios de análises do squid
pflogsumm	V:1.5, I:3.5	173	resumidor de entradas do relatório do Postfix
fwlogwatch	V:0.10, I:0.16	487	analisador de log da firewall
squidview	V:0.04, I:0.41	224	monitoriza e analisa ficheiros access.log do squid
swatch	V:0.07, I:0.30	99	visualizador de ficheiros de registo com correspondência de expressões regulares, destaque e hooks
crm114	V:0.06, I:0.32	1371	Mutilador de Expressões Regulares Controlável e Filtro de Spam (CRM114)
icmpinfo	V:0.01, I:0.35	42	interpretar mensagens ICMP

Tabela 9.4: Lista de analisadores de log do sistema

9.3.3 Amostragem personalizada de dados em texto

Apesar de ferramentas paginadoras com o more(1) e less(1) (veja Secção 1.4.5) e ferramentas personalizadas para destaque e formatação (veja Secção 11.1.8) poderem mostrar dados de texto muito bem, os editores de objetivos gerais (veja Secção 1.4.6) são mais versáteis e personalizáveis.

Dica
Para o vim(1) e o aliás de modo paginador dele view(1), "`:set hls`" ativa pesquisas destacadas.

9.3.4 Amostragem personalizada de hora e data

O formato de apresentação predefinido da hora e da data pelo comando "`ls -l`" depende do **configuração regional** (consulte Secção 1.2.6 para obter o valor). A variável "`$LANG`" é referida em primeiro lugar e pode ser substituída pelas variáveis de ambiente exportadas "`$LC_TIME`" ou "`$LC_ALL`".

O atual formato de apresentação predefinido para cada configuração regional depende da versão da biblioteca C standard (o pacote `libc6`) utilizada. Isto é, diferentes lançamentos Debian têm diferentes predefinições. Para formatos iso, veja ISO 8601.

Se deseja realmente personalizar este formato de amostragem da hora e data para além do **locale**, deve definir o **valor de estilo de hora** com o argumento "`--time-style`" ou com o valor "`$TIME_STYLE`" (veja ls(1), date(1), "`info coreutils 'ls invocation'`").

Dica
Pode eliminar a digitação de uma opção longa na linha de comando utilizando um alias de comando (ver Secção 1.5.9):

```
alias ls='ls --time-style=+%d.%m.%y %H:%M'
```

valor do estilo de hora	localização	mostra a hora e data
`iso`	qualquer	`01-19 00:15`
`long-iso`	qualquer	`2009-01-19 00:15`
`full-iso`	qualquer	`2009-01-19 00:15:16.000000000 +0900`
`locale`	`C`	`Jan 19 00:15`
`locale`	`en_US.UTF-8`	`Jan 19 00:15`
`locale`	`es_ES.UTF-8`	`ene 19 00:15`
`+%d.%m.%y %H:%M`	qualquer	`19.01.09 00:15`
`+%d.%b.%y %H:%M`	`C` ou `en_US.UTF-8`	`19.Jan.09 00:15`
`+%d.%b.%y %H:%M`	`es_ES.UTF-8`	`19.ene.09 00:15`

Tabela 9.5: Mostrar exemplos de hora e data para o comando "`ls -l`" com o valor de **estilo de hora**

9.3.5 Echo de shell colorido

A escrita da shell nos terminais mais modernos pode ser colorida a usar código de escape de ANSI (veja "`/usr/share/doc/`

Por exemplo, tente o seguinte:

```
$ RED=$(printf "\x1b[31m")
$ NORMAL=$(printf "\x1b[0m")
$ REVERSE=$(printf "\x1b[7m")
$ echo "${RED}RED-TEXT${NORMAL} ${REVERSE}REVERSE-TEXT${NORMAL}"
```

9.3.6 Comandos coloridos

Comandos coloridos são úteis para inspeccionar os seus resultados no ambiente interativo. Incluí o seguinte no meu "`~/.bashrc`".

```
if [ "$TERM" != "dumb" ]; then
    eval "`dircolors -b`"
    alias ls='ls --color=always'
    alias ll='ls --color=always -l'
    alias la='ls --color=always -A'
    alias less='less -R'
    alias ls='ls --color=always'
    alias grep='grep --color=always'
    alias egrep='egrep --color=always'
    alias fgrep='fgrep --color=always'
    alias zgrep='zgrep --color=always'
else
    alias ll='ls -l'
    alias la='ls -A'
fi
```

O uso de alias limita os efeitos coloridos da utilização interactiva do comando. Tem vantagem sobre exportar a variável de ambiente "`export GREP_OPTIONS='--color=auto'`" porque a cor pode ser vista sob programas paginadores como o less(1). Se deseja suprimir a cor quando canaliza (pipe) para outros programas, use antes "`--color=auto`" no exemplo em cima para "`~/.bashrc`".

Dica
Pode desligar estes nomes alternativos coloridos no ambiente interativo ao invocar a shell com "`TERM=dumb bash`".

9.3.7 Recordar as atividades do editor para repetições complexas

Pode recordar as atividades do editor para repetições complexas.

Para o Vim, como a seguir.

- ”qa”: inicia a gravação de caracteres teclados no registo nomeado ”a”.
- ... atividades do editor
- ”q”: termina a gravação de caracteres escritos.
- ”@a”: executa o conteúdo do registo ”a”.

Para Emacs, como a seguir.

- ”`C-x (`”: começa a definir uma macro de teclado.
- ... atividades do editor
- ”`C-x )`”: termina de definir uma macro de teclado.
- ”`C-x e`”: executa uma macro de teclado.

9.3.8 Gravar a imagem gráfica de uma aplicação X

Existem algumas maneiras de gravar a imagem gráfica de uma aplicação X, a incluir um ecrã `xterm`.

pacote	popcon	tamanho	ecrã
`gnome-screenshot`	V:12, I:94	1115	Wayland
`flameshot`	V:7, I:17	3542	Wayland
`gimp`	V:34, I:211	33315	Wayland + X
`x11-apps`	V:31, I:443	2461	X
`imagemagick`	V:7, I:272	80	X
`scrot`	V:4, I:48	144	X

Tabela 9.6: Lista de ferramentas gráficas de manipulação de imagens

9.3.9 Gravar alterações em ficheiros de configuração

Existem ferramentas especializadas para registar alterações nos ficheiros de configuração com a ajuda do DVCS e para criar instantâneos do sistema no Btrfs.

pacote	popcon	tamanho	descrição
`etckeeper`	V:24, I:27	157	armazenar ficheiros de configuração e os seus metadados com o Git (predefinição), Mercurial ou GNU Bazaar
`timeshift`	V:7, I:13	4481	utilitário de restauração do sistema usando rsync ou instantâneos BTRFS
`snapper`	V:6.1, I:8.3	2396	Ferramenta de gestão de instantâneos do sistema de ficheiros Linux

Tabela 9.7: Lista de pacotes que podem registar o histórico de configuração

Também pode pensar em usar scripts locais Secção 10.2.3.

9.4 Monitorizar, controlar e iniciar as atividades de programas

As atividades de programas podem ser monitorizadas e controladas a usar ferramentas especiais.

pacote	popcon	tamanho	descrição
coreutils	V:912, I:1000	17994	nice(1): correr um programa com prioridade de agendamento modificada
bsdutils	V:421, I:999	338	renice(1): modifica a prioridade de agendamento de um processo em execução
procps	V:831, I:997	2691	"/proc" utilitários de sistema de ficheiros: ps(1), top(1), kill(1) , watch(1), …
psmisc	V:400, I:715	950	"/proc" utilitários de sistema de ficheiros: killall(1), fuser(1), peekfd(1), pstree(1)
time	V:6, I:76	148	time(1): corre um programa para reportar as utilizações de recursos do sistema no que respeita a tempo
sysstat	V:115, I:164	1904	sar(1), iostat(1), mpstat(1), …: ferramentas de performance do sistema para Linux
isag	V:0.1, I:3.1	109	Interactive System Activity Grapher para sysstat
lsof	V:456, I:950	492	lsof(8): lista os ficheiro abertos por um processo em execução a usar a opção "-p"
strace	V:10, I:100	3209	strace(1): rastreia chamadas e sinais do sistema
ltrace	V:1, I:11	420	ltrace(1): rastreia chamadas de bibliotecas
xtrace	V:0.03, I:0.69	342	xtrace(1): rastreia a comunicação entre cliente X11 e servidor
powertop	V:36, I:230	696	powertop(1): informação sobre a utilização do sistema de energia
cron	V:923, I:997	248	corre processos de acordo com uma agenda nos bastidores a partir do daemon cron(8)
anacron	V:423, I:481	110	agenda de comandos tipo cron para sistemas que não funcionam 24 horas por dia
at	V:69, I:91	158	at(1) ou batch(1): executam um trabalho a uma hora especificada ou abaixo de um certo nível de carga

Tabela 9.8: Lista de ferramentas para monitorizar e controlar as atividades de programas

Dica
Os pacotes procps disponibilizam as bases de monitorizar, controlar e iniciar atividades de programas. Deve aprendê-las todas.

9.4.1 Temporizar um processo

Mostrar o tempo usado pelo processo invocado pelo comando.

```
# time some_command >/dev/null
real    0m0.035s        # time on wall clock (elapsed real time)
user    0m0.000s        # time in user mode
sys     0m0.020s        # time in kernel mode
```

9.4.2 A prioridade de agendamento

Um valor nice é usado para controlar a prioridade de agendamento para o processo.

o valor nice	prioridade de agendamento
19	processo de prioridade menor (nice)
0	processo de prioridade muito alta para o utilizador
-20	processo de prioridade muito alta para o root (não-nice)

Tabela 9.9: Lista de valores nice para a prioridade de agendamento

```
# nice  -19 top                                    # very nice
# nice --20 wodim -v -eject speed=2 dev=0,0 disk.img # very fast
```

Por vezes um valor nice extremo faz mais danos que benefícios ao sistema. Use este comando com cuidado.

9.4.3 O comando ps

O comando ps(1) num sistema Debian suporta ambas funcionalidades do BSD e SystemV e ajuda-o a identificar estaticamente a atividade do processo.

estilo	comando típico	funcionalidade
BSD	`ps aux`	mostrar %CPU %MEM
System V	`ps -efH`	mostra PPID

Tabela 9.10: Lista dos estilos do comando ps

Para o processo filho zombie (defunto), pode matá-lo pelo ID do processo pai identificado no campo ”`PPID`”.

O comando pstree(1) mostra uma árvore de processos.

9.4.4 O comando top

O top(1) no sistema Debian tem funcionalidades ricas e ajuda-o a identificar dinamicamente que processo está a actuar de modo esquisito.

É um programa interativo de ecrã total. Pode obter a ajuda de utilização dele ao pressionar a tecla ”h” e terminá-lo ao pressionar a tecla ”q”.

9.4.5 Listar ficheiros abertos por um processo

Pode listar todos os ficheiros abertos por um processo com o ID do processo (PID), ex. 1, com o seguinte.

```
$ sudo lsof -p 1
```

PID=1 é geralmente o programa de `init`.

9.4.6 Rastear as atividades de programas

Pode rastrear a atividade do programa com o strace(1), ltrace(1), ou xtrace(1) para chamadas de sistema e sinais, chamadas de bibliotecas, ou comunicação entre cliente e servidor do X11.

Pode rastrear as chamadas do sistema do comando `ls` como a seguir.

```
$ sudo strace ls
```

Dica
Utilize o script **strace-graph** que se encontra em **/usr/share/doc/strace/examples/** para fazer uma boa visualização em árvore

9.4.7 Identificação de um processo a usar ficheiros ou sockets

Também pode identificar processos que usam ficheiros pelo fuser(1), ex. para "`/var/log/mail.log`" com o seguinte.

```
$ sudo fuser -v /var/log/mail.log
                     USER        PID ACCESS COMMAND
/var/log/mail.log:   root       2946 F.... rsyslogd
```

Vê que o ficheiro "`/var/log/mail.log`" está aberto para escrita pelo comando rsyslogd(8).

Também pode identificar processos que usam sockets pelo fuser(1), ex. para "`smtp/tcp`" com o seguinte.

```
$ sudo fuser -v smtp/tcp
                     USER        PID ACCESS COMMAND
smtp/tcp:            Debian-exim   3379 F.... exim4
```

Agora sabe que o seu sistema executa o exim4(8) para lidar com as ligações TCP para a porta SMTP (25).

9.4.8 Repetir um comando com um intervalo constante

O watch(1) executa um programa repetidamente num intervalo constante enquanto mostra os seus resultados em ecrã completo.

```
$ watch w
```

Isto mostra quem tem sessão iniciada (logged) no sistema e é atualizado a cada 2 segundos.

9.4.9 Repetir um ciclo de comandos sobre ficheiros

Existem várias maneiras de repetir um ciclo de comandos sobre ficheiros que correspondem a alguma condição, ex. que correspondem ao modelo glob "`*.ext`".

- Método for-loop da shell (veja Secção 12.1.4):

```
for x in *.ext; do if [ -f "$x"]; then command "$x" ; fi; done
```

- combinação do find(1) e do xargs(1):

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | xargs -0 -n 1 command
```

- find(1) com a opção "`-exec`" com um comando:

```
find . -type f -maxdepth 1 -name '*.ext' -exec command '{}' \;
```

- find(1) com a opção "`-exec`" com um script de shell curto:

```
find . -type f -maxdepth 1 -name '*.ext' -exec sh -c "command '{}' && echo 'successful'" \;
```

Os exemplos em cima foram escritos para assegurar o lidar apropriado dos nomes de ficheiros esquisitos como os que contêm espaços. Veja Secção 10.1.5 para utilizações mais avançadas do find(1).

9.4.10 Arrancar um programa a partir da GUI

Para a interface de linha de comandos (CLI), é executado o primeiro programa com o nome correspondente encontrado nos diretórios especificados na variável de ambiente `$PATH`. Veja Secção 1.5.3.

Para a interface gráfica do utilizador (GUI) compatível com as normas freedesktop.org, os ficheiros `*.desktop` no diretório `/usr/share/applications/` fornecem os atributos necessários para a apresentação do menu GUI de cada programa. Cada pacote que é compatível com o sistema de menu xdg do Freedesktop.org instala os seus dados de menu fornecidos por "*.desktop" em "/usr/share/applications/". Os ambientes de trabalho modernos que são compatíveis com a norma Freedesktop.org utilizam estes dados para gerar o seu menu utilizando o pacote xdg-utils. Veja "/usr/share/doc/xdg-utils/README".

Por exemplo, os atributos do ficheiro `chromium.desktop` para o "Navegador Web Chromium" tais como "Nome" para o nome do programa, "Exec" para o caminho de execução do programa e argumentos, "Icon" para o ícone usado, etc. (veja Desktop Entry Specification) como a seguir:

```
[Desktop Entry]
Version=1.0
Name=Chromium Web Browser
GenericName=Web Browser
Comment=Access the Internet
Comment[fr]=Explorer le Web
Exec=/usr/bin/chromium %U
Terminal=false
X-MultipleArgs=false
Type=Application
Icon=chromium
Categories=Network;WebBrowser;
MimeType=text/html;text/xml;application/xhtml_xml;x-scheme-handler/http;x-scheme-handler/ ↩
    https;
StartupWMClass=Chromium
StartupNotify=true
```

Esta é uma descrição muito simplificada. Os ficheiros `*.desktop` são examinados como a seguir:

O ambiente de trabalho define as variáveis de ambiente `$XDG_DATA_HOME` e `$XDG_DATA_DIR`. Por exemplo, sob o GNOME:

- `$XDG_DATA_HOME` é desconfigurada. (É usado o valor predefinido de `$HOME/.local/share`.)
- `$XDG_DATA_DIRS` é definida para `/usr/share/gnome:/usr/local/share/:/usr/share/`.

Para que os diretórios base (veja Especificação de Diretório Base XDG) e os diretórios `applications` sejam como a seguir:

- `$HOME/.local/share/` → `$HOME/.local/share/applications/`
- `/usr/share/gnome/` → `/usr/share/gnome/applications/`
- `/usr/local/share/` → `/usr/local/share/applications/`
- `/usr/share/` → `/usr/share/applications/`

Os ficheiros `*.desktop` são examinados nestes diretórios `applications` por esta ordem.

Dica
Pode ser criada uma entrada personalizada no menu da GUI ao adicionar um ficheiro `*.desktop` no diretório `$HOME/.local/share/applications/`.

Dica
A linha "`Exec=...`" não é analisada pela shell. Utilize o comando env(1) se for necessário definir variáveis de ambiente.

Dica
Se modo semelhante, se um ficheiro `*.desktop` for criado no diretório `autostart` sob esses diretórios base, o programa especificado no ficheiro `*.desktop` é executado automaticamente quando o ambiente de trabalho é iniciado. Veja Especificação de Arranque Automático de Aplicações do Ambiente de Trabalho.

Dica
De modo semelhante, se um ficheiro `*.desktop` for criado no diretório `$HOME/Desktop` e o ambiente de trabalho estiver configurado para suportar funcionalidade de lançamento por ícones do ambiente de trabalho, o programa especificado nele é executado ao se clicar no ícone. Por favor note que o nome real do diretório `$HOME/Desktop` é dependente da localização. Veja xdg-user-dirs-update(1).

9.4.11 Personalizar o programa a ser iniciado

Alguns programas iniciam outros programas automaticamente. Aqui estão alguns pontos de controle para personalizar este processo.

- Menu de configuração da aplicação:
  - Ambiente GNOME: "Definições" → "Aplicações" → "Aplicações Predefinidas"
  - Ambiente KDE: "Lançador de Aplicações" → "Definições do Sistema" → "Aplicações Predefinidas"
  - Navegador Iceweasel: "Editar" → "Preferências" → "Aplicações"
  - mc(1): "`/etc/mc/mc.ext`"
- Variáveis de ambiente como a "`$BROWSER`", "`$EDITOR`", "`$VISUAL`", and "`$PAGER`" (see environ(7))
- O sistema update-alternatives(1) para programas como o "`editor`", "`view`", "`x-www-browser`", "`gnome-www-browser`" e "`www-browser`" (veja Secção 1.4.7)
- os conteúdos dos ficheiros "`~/.mailcap`" e "`/etc/mailcap`" com a associação do tipo MIME com o programa (veja mailcap(5))
- Os conteúdos dos ficheiros "`~/.mime.types`" e "`/etc/mime.types`" que associam a extensão do nome do ficheiro com o tipo MIME (veja run-mailcap(1))

Dica
update-mime(8) atualiza o ficheiro "`/etc/mailcap`" a usar o ficheiro "`/etc/mailcap.order`" (veja mailcap.order(5)).

Dica
O pacote `debianutils` disponibiliza sensible-browser(1), sensible-editor(1) e sensible-pager(1) que fazem decisões sensíveis sobre qual editor, paginador e explorador web chamar, respectivamente. Recomendo-lhe a leitura destes scripts de shell.

Dica
De modo a correr uma aplicação de consola como o `mutt` sob o GUI como a sua aplicação preferida, deve criar uma aplicação GUI como a seguir e definir "`/usr/local/bin/mutt-term`" como a sua aplicação preferida a ser iniciada como descrito.

```
# cat /usr/local/bin/mutt-term <<EOF
#!/bin/sh
gnome-terminal -e "mutt \$@"
EOF
# chmod 755 /usr/local/bin/mutt-term
```

9.4.12 Matar um processo

Use kill(1) para matar (ou enviar um sinal para) um processo pelo ID do processo.

Use killall(1) ou pkill(1) para fazer o mesmo pelo nome do comando do processo ou outro atributo.

valor do sinal	nome do sinal	acção	nota
0	---	nenhum sinal é enviado (ver kill(2))	verificar se o processo está em execução
1	SIGHUP	Terminar o processo	terminal desligado (sinal perdido)
2	SIGINT	Terminar o processo	interrupção a partir do teclado (`CTRL-C`)
3	SIGQUIT	terminar o processo e despejar o núcleo	sair a partir do teclado (`CTRL-\`)
9	SIGKILL	Terminar o processo	sinal de paragem forçada não bloqueável
15	SIGTERM	Terminar o processo	sinal de término bloqueável

Tabela 9.11: Lista dos sinais frequentemente usados para o comando kill

9.4.13 Agendar tarefas uma vez

Corra o comando at(1) para agendar uma tarefa de uma-vez com o seguinte.

```
$ echo 'command -args'| at 3:40 monday
```

9.4.14 Agendar tarefas regularmente

Use cron(8) para agendar tarefas regularmente. Veja crontab(1) e crontab(5).

Pode agendar a execução de processos como um utilizador normal, ex. `foo` ao criar um ficheiro crontab(5) como "`/var/spool/cron/crontabs/foo`" com o comando "`crontab -e`".

Aqui está um exemplo de um ficheiro crontab(5).

```
# use /usr/bin/sh to run commands, no matter what /etc/passwd says
SHELL=/bin/sh
# mail any output to paul, no matter whose crontab this is
MAILTO=paul
# Min Hour DayOfMonth Month DayOfWeek command (Day... are OR'ed)
# run at 00:05, every day
5  0  *  * *   $HOME/bin/daily.job >> $HOME/tmp/out 2>&1
# run at 14:15 on the first of every month -- output mailed to paul
15 14 1  * *   $HOME/bin/monthly
# run at 22:00 on weekdays(1-5), annoy Joe. % for newline, last % for cc:
0 22 *   * 1-5 mail -s "It's 10pm" joe%Joe,%%Where are your kids?%.%%
23 */2 1 2 *   echo "run 23 minutes after 0am, 2am, 4am ..., on Feb 1"
5  4 *   * sun echo "run at 04:05 every Sunday"
# run at 03:40 on the first Monday of each month
40 3 1-7 * *   [ "$(date +%a)" == "Mon" ] && command -args
```

Dica
Para o sistema que não corre continuamente, instale o pacote `anacron` para agendar comandos periódicos a intervalos especificados o mais próximo que os tempos de ligação de máquina permitem. Veja anacron(8) e anacrontab(5).

Dica
Para scripts agendados de manutenção do sistema, pode executá-los periodicamente a partir da conta root ao pôr tais scripts em "`/etc/cron.hourly/`", "`/etc/cron.daily/`", "`/etc/cron.weekly/`", ou "`/etc/cron.monthly/`". Os tempos de execução destes scripts podem ser personalizados pelo "`/etc/crontab`" e "`/etc/anacrontab`".

O Systemd tem capacidade de baixo nível para agendar programas para correr sem o daemon `cron`. Por exemplo, `/lib/systemd/system/apt-daily.timer` e `/lib/systemd/system/apt-daily.service` configuram actividades diárias de download do apt. Veja systemd.timer(5) .

9.4.15 Programação de tarefas em eventos

O Systemd pode agendar programas não só em eventos temporais, mas também em eventos de montagem. Veja Secção 10.2.3.3 e Secção 10.2.3.2 para exemplos.

9.4.16 Tecla Alt-SysRq

Premir Alt-SysRq (PrtScr) seguido de uma tecla faz a magia de recuperar o controlo do sistema.

Veja mais em Guia do utilizador e do administrador do kernel Linux » Linux Magic System Request Key Hacks

tecla que segue a Alt-SysRq	descrição da acção
k	**k**ill (mata) todos os processos na **consola virtual atual** (SAK)
s	**s**incroniza todos os sistemas de ficheiros montados para evitar corrupção de dados
u	remontar todos os sistemas de ficheiros montados em modo de apenas-leitura (**u**mount)
r	restaura o teclado a partir do modo **r**aw após colapso do X

Tabela 9.12: Lista de teclas de comando SAK notáveis

Dica
A partir de um terminal SSH etc., pode usar a funcionalidade Alt-SysRq ao escrever para o "`/proc/sysrq-trigger`". Por exemplo, "`echo s > /proc/sysrq-trigger; echo u > /proc/sysrq-trigger`" a partir do aviso da shell de root **s**sincroniza e **u**mounts (desmonta) todos os sistemas de ficheiros montados.

O kernel Linux Debian amd64 atual (2021) tem `/proc/sys/kernel/sysrq=438=0b110110110`:

- 2 = 0x2 - permite o controlo do nível de registo da consola (ON)
- 4 = 0x4 - ativar o controlo do teclado (SAK, unraw) (ON)
- 8 = 0x8 - ativar depuração de processos, etc. (OFF)
- 16 = 0x10 - ativar comando de sincronização (ON)
- 32 = 0x20 - ativar remontagem só de leitura (ON)
- 64 = 0x40 - ativa a sinalização de processos (term, kill, oom-kill) (OFF)
- 128 = 0x80 - permitir o reinício/desligar(ON)
- 256 = 0x100 - permite iniciar todas as tarefas RT (ON)

9.5 Dicas de manutenção do sistema

9.5.1 Quem está no sistema?

Pode verificar quem está no sistema com o seguinte.

- who(1) mostra quem tem sessão iniciada.
- w(1) mostra quem tem sessão iniciada e o que estão a fazer.
- last(1) mostra a listagem do último utilizador a iniciar sessão.
- `lastb`(1)mostra a listagem dos últimos utilizadores a falharem o inicio de sessão.

Dica
"`/var/run/utmp`" e "`/var/log/wtmp`" detém tal informação do utilizador. Veja login(1) e utmp(5).

9.5.2 Avisar todos

Pode mandar uma mensagem para todos os que têm sessão iniciada no sistema com wall(1) com o seguinte.

```
$ echo "We are shutting down in 1 hour" | wall
```

9.5.3 Identificação do hardware

Para os dispositivos tipo PCI (AGP, PCI-Express, CardBus, ExpressCard, etc.), o lspci(8) (provavelmente com a opção "`-nn`") é um bom inicio para a identificação do hardware.

Alternativamente, pode identificar o hardware ao ler os conteúdos de "`/proc/bus/pci/devices`" ou explorar a árvore de diretórios sob "`/sys/bus/pci`" (veja Secção 1.2.12).

pacote	popcon	tamanho	descrição
`pciutils`	V:264, I:994	289	Utilitários PCI do Linux: lspci(8)
`usbutils`	V:90, I:892	322	Utilitários USB do Linux: lsusb(8)
`nvme-cli`	V:27, I:36	2222	Utilitários NVMe para Linux: nvme(1)
`pcmciautils`	V:4.1, I:6.1	92	Utilitários PCMCIA par Linux: pccardctl(8)
`scsitools`	V:0.2, I:2.1	261	colecção de ferramentas para gestão de hardware SCSI: lsscsi(8)
`procinfo`	V:0.4, I:5.8	149	informação do sistema obtida de "`/proc`": lsdev(8)
`lshw`	V:12, I:88	971	informação acerca da configuração do hardware: lshw(1)
`discover`	V:24, I:561	81	sistema de identificação de hardware: discover(8)

Tabela 9.13: Lista de ferramenta de identificação de hardware

9.5.4 Configuração do hardware

Apesar da maioria da configuração de hardware nos sistemas de ambiente de trabalho GUI modernos como o GNOME e KDE poder ser gerida através de acompanhamento por ferramentas de configuração com GUI, é uma boa ideia conhecer alguns métodos básicos de o configurar.

Para a configuração do teclado e do terminal, veja Secção 8.2 e Secção 8.3.

Aqui, o ACPI é uma estrutura mais recente para o sistema de gestão de energia que o APM.

Dica
O escalar de frequências da CPU em sistemas modernos é governado por módulos do kernel como o `acpi_cpufreq`.

9.5.5 Hora do sistema e do hardware

O seguinte define a hora do sistema e hardware para MM/DD hh:mm, AAAA.

```
# date MMDDhhmmCCYY
# hwclock --utc --systohc
# hwclock --show
```

A horas são mostradas normalmente na hora local no sistema Debian mas o hardware e a hora do sistema geralmente usam UTC(GMT).

Se a hora do hardware estiver definida para UTC, mude a definição para "`UTC=yes`" em "`/etc/default/rcS`".

O seguinte reconfigura a zona horária usada pelo sistema Debian.

pacote	popcon	tamanho	descrição
console-setup	V:81, I:976	422	tipo de letra da consola Linux e utilitários da tabela de teclas
keyd	V:1.5, I:1.7	1437	Daemon de remapeamento de teclas do teclado que usa nível de kernel de entrada primitiva (evdev, uinput)
keyd-application-mapper	V:0.01, I:0.08	34	Daemon de remapeamento de teclas do teclado - remapeador específico da aplicação
x11-xserver-utils	V:373, I:533	559	Utilitários do servidor X: xset(1), xmodmap(1)
acpid	V:52, I:78	158	daemon para gerir eventos entregues pelo Advanced Configuration and Power Interface (ACPI)
acpi	V:6, I:73	49	utilitário para mostrar informação em dispositivos ACPI
sleepd	V:0.07, I:0.10	84	daemon para pôr um portátil em modo de adormecimento durante a inatividade
hdparm	V:110, I:200	246	optimização do acesso ao disco rígido (veja Secção 9.6.9)
smartmontools	V:242, I:270	2455	controlar e monitorizar sistemas de armazenamento a usar S.M.A.R.T.
setserial	V:3.0, I:4.9	104	colecção de ferramentas para gestão de portas série
memtest86+	V:1, I:19	12483	colecção de ferramentas para gestão de hardware de memória
scsitools	V:0.2, I:2.1	261	colecção de ferramentas para gestão de hardware SCSI
setcd	V:0.05, I:0.61	33	optimização de acesso a drives de discos compactos
big-cursor	I:0.67	26	cursores de rato maiores para o X

Tabela 9.14: Lista de ferramentas de configuração do hardware

```
# dpkg-reconfigure tzdata
```

Se desejar atualizar a hora do sistema através da rede, considere usar o serviço NTP como pacotes como os `ntp`, `ntpdate` e `chrony`.

Dica
Sob systemd, use `systemd-timesyncd` para a sincronização da hora com a rede. Veja systemd-timesyncd(8).

Veja o seguinte.

- Como Gerir a Data e Hora com Precisão
- NTP Projeto de Serviços Públicos
- O pacote `ntp-doc`

Dica
O ntptrace(8) no pacote `ntp` pode rastrear uma cadeia de servidores NTP até à sua fonte principal.

9.5.6 A infraestrutura de som

As drivers para placas de som para o Linux atual são disponibilizadas pelo Advanced Linux Sound Architecture (ALSA). ALSA disponibiliza um modo de emulação para o anterior Open Sound System (OSS) para compatibilidade.

A aplicação de software pode ser configurada não apenas para aceder a dispositivos de som diretamente, mas também para aceder-lhes através de algum sistema de servidor de som normalizado. Atualmente, PulseAudio,

JACK, e PipeWire são usados como sistemas de servidores de som. Veja a página wiki do Debian sobre Som para a situação mais recente.

Existe normalmente um motor de som comum para cada ambiente de trabalho popular. Cada motor de som usado pela aplicação pode escolher ligar a diferentes servidores de som.

Dica
Use "`cat /dev/urandom > /dev/audio`" ou speaker-test(1) para testar os altifalantes (^C para parar).

Dica
Se não conseguir obter som, os seus altifalantes podem estar ligados a uma saída silenciada (mute). Os sistemas de som modernos têm muitas saídas. O alsamixer(1) no pacote `alsa-utils` é útil para configurar as definições de volume e mute.

pacote	popcon	tamanho	descrição
`alsa-utils`	V:342, I:460	2700	utilitários para configurar e utilizar ALSA
`oss-compat`	V:0.6, I:9.2	21	compatibilidade de OSS sob ALSA para prevenir erros de "`/dev/dsp não encontrado`"
`pipewire`	V:326, I:370	136	motor de processamento de áudio e vídeo servidor multimédia - meta-pacote
`pipewire-bin`	V:333, I:370	2251	motor de processamento de áudio e vídeo servidor multimédia - servidor de áudio e programas CLI
`pipewire-alsa`	V:172, I:241	194	motor de processamento de áudio e vídeo servidor multimédia - servidor de áudio para substituir o ALSA
`pipewire-pulse`	V:298, I:347	53	motor de processamento de áudio e vídeo servidor multimédia - servidor de áudio para substituir o PulseAudio
`pulseaudio`	V:147, I:171	6616	servidor PulseAudio
`libpulse0`	V:443, I:579	984	biblioteca cliente PulseAudio
`jackd`	V:2, I:14	8	servidor (baixa latência) JACK Audio Connection Kit. (JACK)
`libjack0`	V:1.8, I:8.6	328	biblioteca (baixa latência) JACK Audio Connection Kit. (JACK)
`libphonon4qt5-4`	V:18, I:42	572	Phonon: motor de som do KDE

Tabela 9.15: Lista de pacotes de som

9.5.7 desativar o protector de ecrã (screensaver)

Para desativar o protector de ecrã, utilize os seguintes comandos.

ambiente	comando
A consola do Linux	`setterm -powersave off`
O X Window (desativar o protector de ecrã)	`xset s off`
O X Window (desativar o dpms)	`xset -dpms`
O X Window (GUI de configuração do protector de ecrã)	`xscreensaver-command -prefs`

Tabela 9.16: Lista de comandos para desativar o protector de ecrã

9.5.8 desativar os sons de beep

Pode-se sempre desligar o altifalante do PC para desativar os apitos. Remover o módulo de kernel `pcspkr` faz isso por si.

O seguinte previne o programa readline(3) usado pelo bash(1) de apitar quando encontra um caractere de alerta (ASCII=7).

```
$ echo "set bell-style none">> ~/.inputrc
```

9.5.9 Utilização da memória

Existem 2 recursos disponíveis para obter o estado da utilização de memória.

- A mensagem de arranque do kernel em "`/var/log/dmesg`" contém o tamanho total exacto da memória disponível.
- free(1) e top(1) mostram informação sobre os recursos de memória no sistema em execução.

Aqui está um exemplo.

```
# grep '\] Memory' /var/log/dmesg
[    0.004000] Memory: 990528k/1016784k available (1975k kernel code, 25868k reserved, 931k ↩
    data, 296k init)
$ free -k
             total       used       free     shared    buffers     cached
Mem:        997184     976928      20256          0     129592     171932
-/+ buffers/cache:     675404     321780
Swap:      4545576          4    4545572
```

Pode estar a pensar "o dmesg fala-me em 990 MB livres e o free -k diz 320 MB livres. Faltam mais de 600 MB ...".

Não se preocupe com o grande tamanho de "`used`" e o pequeno tamanho de "`free`" na linha "`Mem:`", mas leia a que está sob elas (675404 e 321780 no exemplo em cima) e relaxe.

Para o meu MacBook com 1GB=1048576k de DRAM (o sistema de vídeo rouba alguma), vejo o seguinte.

relatório	**tamanho**
Tamanho total no dmesg	1016784k = 1GB - 31792k
Livre no dmesg	990528k
Total sob a shell	997184k
Livre sob a shell	20256k (mas efectivamente 321780k)

Tabela 9.17: Lista dos tamanhos de memória reportados

9.5.10 Segurança do sistema e verificação de integridade

Uma manutenção pobre do sistema pode expor o seu sistema à exploração externa.

Para segurança do sistema e verificação de integridade, deve começar com o seguinte.

- O pacote `debsums`, veja debsums(1) e Secção 2.5.2.
- O pacote `chkrootkit`, veja chkrootkit(1).
- A família de pacotes `clamav`, veja clamscan(1) e freshclam(1).
- FAQ de Segurança Debian.

pacote	popcon	tamanho	descrição
logcheck	V:4.4, I:5.4	120	daemon para enviar as anomalias nos ficheiros de log do sistema ao administrador por mail
debsums	V:4, I:29	108	utilitário para verificar os pacotes instalados contra chaves de verificação MD5
chkrootkit	V:10, I:14	987	detector de rootkit
clamav	V:8, I:38	19877	utilitário de anti-vírus para Unix - interface de linha de comandos
tiger	V:1.0, I:1.3	7800	relatar vulnerabilidades de segurança do sistema
tripwire	V:1.1, I:1.4	5060	verificador de integridade de ficheiros e diretórios
john	V:1.1, I:7.3	469	ferramenta activa de crack de palavras-passe
aide	V:2.2, I:2.6	328	Ambiente de Detecção de Intrusão Avançado - binário estático
integrit	V:0.09, I:0.17	2759	programa de verificação de integridade de ficheiros
crack	V:0.06, I:0.55	151	programa de adivinhação de palavra-passe

Tabela 9.18: Lista de ferramentas para segurança do sistema e verificação de integridade

- Manual de Segurança Debian.

Aqui está um script simples para verificar as típicas permissões de ficheiros escritas incorrectamente.

```
# find / -perm 777 -a \! -type s -a \! -type l -a \! \( -type d -a -perm 1777 \)
```

Cuidado
Como o pacote `debsums` usa sumários de verificação MD5 armazenados localmente, não pode ser de total confiança como ferramenta de auditoria à segurança do sistema contra ataques maliciosos.

9.6 Dicas de armazenamento de dados

Arrancar o seu sistema coo o sistema Linux live (veja Secção 3.2.2 and Secção 3.2.3) torna mais fácil para si reconfigurar o armazenamento de dados no sistema instalado.

Nota
Declarações sobre disco rijo (HDD) são aplicáveis a outros dispositivos de armazenamento tais como SSD / Pen flash USB / Cartão de memória / ... . Substitua os nomes dos dispositivos nos exemplos como `/dev/sda` com nomes de dispositivos apropriados `/dev/nvme0`, `/dev/mmcblk0`, ...

Poderá ser necessário desmontar(8) alguns dispositivos manualmente a partir da linha de comandos antes de os operar, caso sejam montados automaticamente pelo sistema de ambiente de trabalho GUI.

9.6.1 Utilização do espaço em disco

A utilização do espaço em disco pode ser avaliada por programas disponibilizados pelos pacotes `mount`, `coreutils` e `xdu`:

- mount(8) reporta todos os sistemas de ficheiros montados (= discos).

- df(1) reporta a utilização do espaço em disco para o sistema de ficheiros.
- du(1) reporta a utilização do espaço em disco para a árvore do diretório.

Dica
Pode alimentar a saída de du(8) a xdu(1) para produzir a apresentação gráfica dele e interactiva com "`du -k . |xdu`", "`sudo du -k -x / |xdu`", etc.

9.6.2 Configuração das partições do disco

Para configuração de partições de disco, apesar do fdisk(8) ser considerado o standard, o parted(8) merece alguma atenção. "Dados de particionamento do disco", "Tabela de partições", "Mapa de partições" e "Etiqueta do disco" são todos sinónimos.

PCs antigos usam o esquema clássico do Master Boot Record (MBR) para manter os dados de partições do disco no primeiro sector, isto é, LBA sector 0 (512 bytes).

PCs novos com Unified Extensible Firmware Interface (UEFI), incluindo os Macs baseados em Intel, usam o esquema GUID Partition Table (GPT) para manter os dados de partições do disco não no primeiro sector.

Apesar do fdisk(8) ter sido o standard como ferramenta de particionamento de disco, o parted(8) está a substituí-lo.

pacote	popcon	tamanho	descrição
`util-linux`	V:913, I:1000	4826	vários utilitários de sistema incluindo fdisk(8) e cfdisk(8)
`parted`	V:449, I:569	126	GNU Parted programa de redimensionamento de partições do disco
`gparted`	V:12, I:83	2313	Editor de partições do GNOME baseado na `libparted`
`gdisk`	V:19, I:253	967	editor de partições para o disco híbrido GPT/MBR
`kpartx`	V:15, I:24	78	programa para criar mapeamentos de dispositivo para partições

Tabela 9.19: Lista de pacotes de gestão de partições do disco



Cuidado
Apesar do parted(8) afirmar também criar e redimensionar sistemas de ficheiros, é mais seguro fazer tais coisas a usar ferramentas especializadas e com melhor manutenção como as ferramentas mkfs(8) (mkfs.msdos(8), mkfs.ext2(8), mkfs.ext3(8), mkfs.ext4(8), …) e resize2fs(8).

Nota
De modo a mudar entre GPT e MBR, precisa de apagar os primeiros blocos de conteúdo do disco directamente (veja Secção 9.8.6) e usar "`parted /dev/sdx mklabel gpt`" ou "`parted /dev/sdx mklabel msdos`" para o definir. Por favor note que "`msdos`" é usado aqui para o MBR.

9.6.3 Aceder a partição a usar UUID

Apesar da reconfiguração da sua partição ou ordem de ativação de medias de armazenamento amovíveis poder apresentar nomes diferentes para as partições, pode aceder-lhes de modo consistente. Isto é também útil se tem múltiplos discos e a sua BIOS/UEFI não lhes fornecer nomes de dispositivo consistentes.

- mount(8) com a opção "`-U`" pode montar um dispositivo de bloco a usar o UUID, em vez de usar o nome de ficheiro dele tal como "`/dev/sda3`".
- "`/etc/fstab`" (veja fstab(5)) pode usar UUID.
- Os gestores de arranque (Secção 3.1.2) também podem usar UUID.

Dica
Pode testar o UUID de um dispositivo especial de bloco com blkid(8).
Também pode sondar o UUID e outras informações com "`lsblk -f`".

9.6.4 LVM2

LVM2 é um gestor de volumes lógicos para o kernel Linux. Com o LVM2, podem ser criadas partições de disco em volumes lógicos em vez de discos rijos físicos.

O LVM requer o seguinte.

- suporte a device-mapper no kernel Linux (predefinido para os kernels Debian)
- a biblioteca de suporte a device-mapper no espaço de utilizador (pacote (`libdevmapper*`)
- as ferramentas LVM2 do espaço de utilizador (pacote `lvm2`)

Por favor comece a aprender LVM2 a partir dos seguintes manuais.

- lvm(8): Bases do mecanismo LVM2 (lista de todos os comandos LVM2)
- lvm.conf(5): Ficheiro de configuração para LVM2
- lvs(8): Reporta informação acerca de volumes lógicos
- vgs(8): Reporta informação acerca de grupos de volumes
- pvs(8): Reporta informação acerca de volumes físicos

9.6.5 Configuração do sistema de ficheiros

Para o sistema de ficheiro ext4, o pacote `e2fsprogs` disponibiliza o seguinte.

- mkfs.ext4(8) para criar um novo sistema de ficheiros ext4
- fsck.ext4(8) para verificar e reparar um sistema de ficheiros ext4 existente
- tune2fs(8) para configurar o super-bloco do sistema de ficheiros ext4
- debugfs(8) para depurar um sistema de ficheiros ext4 interativamente. (Era o comando `undel` para recuperar ficheiros apagados.)

Os comandos mkfs(8) e fsck(8) são disponibilizados pelo pacote `e2fsprogs` como frontends para vários programas dependentes do sistema de ficheiros (`mkfs.fstype` e `fsck.fstype`). Para o sistema de ficheiros ext4 existem os mkfs.ext4(8) e o fsck.ext4(8) (estão ligados simbolicamente ao mke2fs(8) and e2fsck(8)).

Estão disponíveis comandos semelhantes para cada sistema de ficheiros suportado pelo Linux.

pacote	popcon	tamanho	descrição
e2fsprogs	V:821, I:997	1560	utilitários para os sistemas de ficheiros ext2/ext3/ext4
btrfs-progs	V:52, I:79	5255	utilitários para o sistema de ficheiros btrfs
reiserfsprogs	V:11, I:21	473	utilitários para o sistema de ficheiros Reiserfs
zfsutils-linux	V:32, I:33	1905	utilitários para o sistema de ficheiros OpenZFS
dosfstools	V:265, I:568	310	utilitários para o sistema de ficheiros FAT. (Microsoft: MS-DOS, Windows)
exfatprogs	V:36, I:466	580	para o sistema de ficheiros exFAT mantido pela Samsung.
exfat-fuse	V:2, I:40	73	controlador de leitura/gravação de sistema de arquivos exFAT (Microsoft) para FUSE.
xfsprogs	V:42, I:84	4651	utilitários para o sistema de ficheiros XFS. (SGI: IRIX)
ntfs-3g	V:208, I:514	1507	controlador de leitura/gravação do sistema de arquivos NTFS (Microsoft: Windows NT, ...) para FUSE.
jfsutils	V:0.4, I:6.7	1104	utilitários para o sistema de ficheiros JFS. (IBM: AIX, OS/2)
xorriso	V:15, I:65	378	utilitários para sistema de ficheiros ISO-9660 e escrita de CD/DVD a partir de libburnia
wodim	V:7, I:83	898	ferramenta de escrita de CD/DVD de linha de comandos a partir de cdrkit
genisoimage	V:17, I:153	1567	ferramenta de sistema de ficheiros ISO-9660 de linha de comandos a partir de cdrkit
reiser4progs	V:0.1, I:1.1	1367	utilitários para o sistema de ficheiros Reiser4
hfsprogs	V:0.3, I:3.1	394	utilitários para os sistemas de ficheiros HFS e HFS Plus. (Apple: Mac OS)
zerofree	V:7, I:116	30	programa para zerar blocos livres de sistemas de ficheiros ext2/3/4

Tabela 9.20: Lista de pacotes de gestão de sistemas de ficheiros

Dica
O sistema de ficheiros Ext4 é o sistema de ficheiros predefinido para o sistema Linux e a utilização é fortemente recomendada a menos que tenha razões especificas para não o fazer.
O estado do Btrfs pode ser encontrado em Debian wiki on btrfs e kernel.org wiki on btrfs. Espera-se que seja o próximo sistema de ficheiros padrão depois do sistema de ficheiros ext4.
Algumas ferramentas permitem acesso a sistemas de ficheiros sem suporte do kernel do Linux (veja Secção 9.8.2).

9.6.6 Criação do sistema de ficheiros e verificação de integridade

O comando mkfs(8) cria o sistema de ficheiros num sistema Linux. O comando fsck(8) disponibiliza a verificação de integridade e reparação do sistema de ficheiros num sistema Linux.

Debian agora, por predefinição, não faz `fsck` periódicos após a criação do sistema de ficheiros.

Cuidado
Geralmente não é seguro correr o `fsck` em **sistemas de ficheiros montados**.

Dica
Pode executar o comando fsck(8) com segurança em todos os sistemas de ficheiros incluindo o sistema raiz durante o arranque da máquina ao definir "`enable_periodic_fsck`" em "`/etc/mke2fs.conf`" e a contagem máxima de montagens para 0 a usar "`tune2fs -c0 /dev/`*`partition_name`*". Veja mke2fs.conf(5) e tune2fs(8). Verifique os ficheiros em "`/var/log/fsck/`" pelos resultados do comando fsck(8) executado do script de arranque.

9.6.7 Optimização do sistema de ficheiros por opções de montagem

A configuração estática básica dos sistemas de ficheiros é dada por "`/etc/fstab`". Por exemplo,

```
«file system»                    «mount point» «type» «options»    «dump» «pass»
proc                                     /proc proc   defaults          0 0
UUID=709cbe4c-80c1-56db-8ab1-dbce3146d2f7 /     ext4   errors=remount-ro 0 1
UUID=817bae6b-45d2-5aca-4d2a-1267ab46ac23 none  swap   sw                0 0
/dev/scd0                         /media/cdrom0  udf,iso9660 user,noauto  0 0
```

Dica
O UUID (veja Secção 9.6.3) pode ser utilizado para identificar um dispositivo de bloco em vez de nomes vulgares de dispositivos de bloco, tal como "`/dev/sda1`", "`/dev/sda2`",...

Desde o Linux 2.6.30, o kernel adota por defeito o comportamento fornecido pela opção "`relatime`".

Veja fstab(5) e mount(8).

9.6.8 Optimização do sistema de ficheiros através do superblock

As características de um sistema de ficheiros podem ser optimizadas via o super-bloco dele a usar o comando tune2fs(8).

- A execução de "`sudo tune2fs -l /dev/sda1`" mostra o conteúdo do super-bloco do sistema de ficheiros em "`/dev/sda1`".
- A execução de "`sudo tune2fs -c 50 /dev/sda1`" muda a frequência das verificações do sistema de ficheiros (execução do `fsck` durante o arranque) para cada 50 arranques em "`/dev/sda1`".
- A execução de "`sudo tune2fs -j /dev/sda1`" adiciona capacidade de journal ao sistema de ficheiros, isto é, conversão de ext2 para ext3 em "`/dev/sda1`". (Faça isto no sistema de ficheiros desmontado.)
- A execução de "`sudo tune2fs -O extents,uninit_bg,dir_index /dev/sda1 && fsck -pf /dev/sda1`" converte-o de ext3 para ext4 em "`/dev/sda1`". (Faça isto no sistema de ficheiros desmontado.)

Dica
Apesar do nome dele, o tune2fs(8) não funciona apenas no sistema de ficheiros ext2, mas também nos sistemas de ficheiros ext3 e ext4.

9.6.9 Optimização do disco rígido



Atenção
Por favor verifique o seu hardware e leia o manual do hdparm(8) antes de brincar com a configuração do disco rígido porque isto pode ser bastante perigoso para a integridade dos dados.

Pode testar a velocidade de acesso ao disco de um disco rígido, p.e. "`/dev/sda`", por "`hdparm -tT /dev/sda`". Para algum disco rígido ligado com (E)IDE, pode acelerá-lo com "`hdparm -q -c3 -d1 -u1 -m16 /dev/sda`" ao ativar o suporte a "(E)IDE 32-bit I/O", a ativar a flag "using_dma", a definir a flag "interrupt-unmask" e a definir o "multiple 16 sector I/O" (perigoso!).

Pode testar a funcionalidade de cache de escrita de um disco rígido, por exemplo "`/dev/sda`", com "`hdparm -W /dev/sda`". Pode desativar a funcionalidade de cache de escrita dele com "`hdparm -W 0 /dev/sda`".

Pode ser capaz de ler CDROMs muito pressionados em drives de CDROM modernas de alta velocidade ao abrandá-la com "`setcd -x 2`".

9.6.10 Optimização de disco de estado sólido (SSD)

A unidade de estado sólido (SSD) é agora automaticamente detetada.

Reduza os acessos desnecessários ao disco para evitar o desgaste do disco, montando "`tmpfs`" no caminho de dados volátil em `/etc/fstab`.

9.6.11 Usar SMART para prever falhas no disco rígido

Pode monitorizar e registar em log o seu disco rígido que é compatível com SMART com o daemon smartd(8).

1. ativar a função SMART na BIOS.
2. Instalar o pacote `smartmontools`.
3. Identificar os seus discos rígidos ao listá-los com df(1).
   - Vamos assumir uma drive de disco rígido a ser monitorizada como "`/dev/sda`".
4. Verifique o resultado de "`smartctl -a /dev/sda`" para ver se a funcionalidade SMART está atualmente ligada.
   - Se não, active-o com "`smartctl -s on -a /dev/sda`".
5. Active o daemon smartd(8) ao correr o seguinte.
   - retire a marca de comentário na linha "`start_smartd=yes`" no ficheiro "`/etc/default/smartmontools`".
   - reiniciar o daemon smartd(8) com "`sudo systemctl restart smartmontools`".

Dica
O daemon smartd(8) pode ser personalizado com o ficheiro `/etc/smartd.conf` incluindo em como ser notificado dos avisos.

9.6.12 Especifique o diretório de armazenamento temporário através de $TMPDIR

As aplicações criam ficheiros temporários normalmente sob o diretório de armazenamento temporário "`/tmp`". Se "`/tmp`" não disponibilizar espaço suficiente, pode especificar um diretório de espaço temporário, a programas bem-comportados, através da variável `$TMPDIR`.

9.6.13 Expandir o espaço de armazenamento utilizável via LVM

Para partições criadas em Logical Volume Manager (LVM) (funcionalidade do Linux) durante a instalação, elas podem ser redimensionadas facilmente ao concatenar extensões nelas ou ao truncar extensões delas sobre múltiplos dispositivos de armazenamento sem grandes reconfigurações do sistema.

9.6.14 Expandir o espaço de armazenamento utilizável ao montar outra partição

Se tiver uma partição vazia (ex. "`/dev/sdx`"), pode formatá-la com mkfs.ext4(1) e mount(8) para um diretório onde precise de mais espaço. (necessita copiar os conteúdos originais.)

```
$ sudo mv work-dir old-dir
$ sudo mkfs.ext4 /dev/sdx
$ sudo mount -t ext4 /dev/sdx work-dir
$ sudo cp -a old-dir/* work-dir
$ sudo rm -rf old-dir
```

Dica
Pode em alternativa montar um ficheiro de imagem de disco vazio (veja Secção 9.7.5) como um dispositivo de loop (veja Secção 9.7.3). A utilização real do disco cresce com os dados reais armazenados.

9.6.15 Expandir o espaço de armazenamento utilizável ao fazer bind-mount para outro diretório

Se tiver um diretório vazio (p.e. "`/caminho/para/diretório-vazio`") com espaço utilizável noutra partição, pode fazer mount(8) ao mesmo com a opção "`--bind` para um diretório (p.e., "`diretório-de-trabalho`") onde necessite de mais espaço.

```
$ sudo mount --bind /path/to/emp-dir work-dir
```

9.6.16 Expansão do espaço de armazenamento utilizável ao fazer overlay-mounting para outro diretório

Se tem espaço utilizável noutra partição (ex. "`/path/to/empty`") e "`/path/to/work`"), pode criar um diretório nela e empilhá-lo no diretório antigo (ex, "`/path/to/old`") onde precisa de espaço a usar o OverlayFS para Linux kernel 3.18 ou mais recente (Debian Stretch 9.0 ou posterior).

```
$ sudo mount -t overlay overlay \
  -olowerdir=/path/to/old-dir,upperdir=/path/to/empty,workdir=/path/to/work
```

Aqui, "`/path/to/empty`" e "`/path/to/work`" devem estar na partição com Escrita-Leitura activa a escrever em "`/path/to/old`".

9.6.17 Expandir o espaço de armazenamento utilizável a usar ligações simbólicas



Cuidado
Este é um método descontinuado. Alguns programas podem não funcionar bem com uma "ligação simbólica a um diretório". Em vez disso, use as opções de "montagem" descritas em cima.

Se tem um diretório vazio (ex. "`/caminho/para/diretório-vazio`") noutra partição com espaço utilizável, pode criar uma ligação simbólica ao diretório com o ln(8).

```
$ sudo mv work-dir old-dir
$ sudo mkdir -p /path/to/emp-dir
$ sudo ln -sf /path/to/emp-dir work-dir
$ sudo cp -a old-dir/* work-dir
$ sudo rm -rf old-dir
```



Atenção
Não utilize uma "ligação simbólica para um diretório" para diretórios geridos pelo sistema, tais como o "`/opt`". Tal ligação simbólica poderá ser sobrescrita quando o sistema for atualizado.

9.7 A imagem de disco

Aqui discutimos manipulações da imagem do disco.

9.7.1 Criar o ficheiro de imagem de disco

O ficheiro de imagem de disco, "`disco.img`", de um dispositivo não montado, ex., a segunda drive SCSI ou serial ATA "`/dev/sdb`", pode ser feito por um dos seguintes.

```
# dd if=/dev/sdb of=disk.img; sync
```

```
# cp /dev/sdb disk.img ; sync
```

```
# cat /dev/sdb > disk.img ; sync
```

O master boot record (MBR) da imagem de disco dos PC's tradicionais (veja Secção 9.6.2) que reside no primeiro sector no disco IDE primário pode ser feito a usar o dd(1) com o seguinte.

```
# dd if=/dev/sda of=mbr.img bs=512 count=1
# dd if=/dev/sda of=mbr-nopart.img bs=446 count=1
# dd if=/dev/sda of=mbr-part.img skip=446 bs=1 count=66
```

- "`mbr.img`": O MBR com a tabela de partições
- "`mbr-nopart.img`": O MBR sem a tabela de partições
- "`mbr-part.img`": A tabela de partições apenas do MBR

Se está a criar uma imagem de uma partição de disco do disco original, substitua "`/dev/sda`" por "`/dev/sda1`" etc.

9.7.2 Escrever directamente no disco

O ficheiro de imagem de disco "`disk.img`" pode ser escrito para um dispositivo desmontado, ex. a segunda drive SCSI "`/dev/sdb`" com tamanho correspondente, por uma das seguintes maneiras.

```
# dd if=disk.img of=/dev/sdb ; sync
```

```
# cp disk.img /dev/sdb ; sync
```

```
# cat disk.img >/dev/sdb ; sync
```

Se modo semelhante, o ficheiro de imagem de partição de disco, ”`partition.img`” pode ser escrito para uma partição desmontada, ex., a primeira partição do segundo disco SCSI ”`/dev/sdb1`” com tamanho correspondente, com o seguinte.

```
# dd if=partition.img of=/dev/sdb1 ; sync
```

9.7.3 Montar o ficheiro de imagem de disco

A imagem de disco ”`partition.img`” que contém uma partição única pode ser montada e desmontada ao usar o dispositivo loop como a seguir.

```
# losetup --show -f partition.img
/dev/loop0
# mkdir -p /mnt/loop0
# mount -t auto /dev/loop0 /mnt/loop0
...hack...hack...hack
# umount /dev/loop0
# losetup -d /dev/loop0
```

Isto pode ser simplificado como a seguir.

```
# mkdir -p /mnt/loop0
# mount -t auto -o loop partition.img /mnt/loop0
...hack...hack...hack
# umount partition.img
```

Cada partição da imagem de disco ”`disk.img`” que contém múltiplas partições pode ser montada a usar o dispositivo loop.

```
# losetup --show -f -P disk.img
/dev/loop0
# ls -l /dev/loop0*
brw-rw---- 1 root disk   7,  0 Apr  2 22:51 /dev/loop0
brw-rw---- 1 root disk 259, 12 Apr  2 22:51 /dev/loop0p1
brw-rw---- 1 root disk 259, 13 Apr  2 22:51 /dev/loop0p14
brw-rw---- 1 root disk 259, 14 Apr  2 22:51 /dev/loop0p15
# fdisk -l /dev/loop0
Disk /dev/loop0: 2 GiB, 2147483648 bytes, 4194304 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 6A1D9E28-C48C-2144-91F7-968B3CBC9BD1

Device         Start     End Sectors  Size Type
/dev/loop0p1  262144 4192255 3930112  1.9G Linux root (x86-64)
/dev/loop0p14   2048    8191    6144    3M BIOS boot
/dev/loop0p15   8192  262143  253952  124M EFI System

Partition table entries are not in disk order.
# mkdir -p /mnt/loop0p1
# mkdir -p /mnt/loop0p15
# mount -t auto /dev/loop0p1 /mnt/loop0p1
# mount -t auto /dev/loop0p15 /mnt/loop0p15
```

```
# mount |grep loop
/dev/loop0p1 on /mnt/loop0p1 type ext4 (rw,relatime)
/dev/loop0p15 on /mnt/loop0p15 type vfat (rw,relatime,fmask=0002,dmask=0002,allow_utime ↩
    =0020,codepage=437,iocharset=ascii,shortname=mixed,utf8,errors=remount-ro)
...hack...hack...hack
# umount /dev/loop0p1
# umount /dev/loop0p15
# losetup -d /dev/loop0
```

Alternativamente, podem-se fazer efeitos semelhantes ao usar os dispositivos device mapper criados pelo kpartx(8) do pacote `kpartx` como a seguir.

```
# kpartx -a -v disk.img
add map loop0p1 (253:0): 0 3930112 linear 7:0 262144
add map loop0p14 (253:1): 0 6144 linear 7:0 2048
add map loop0p15 (253:2): 0 253952 linear 7:0 8192
# fdisk -l /dev/loop0
Disk /dev/loop0: 2 GiB, 2147483648 bytes, 4194304 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 6A1D9E28-C48C-2144-91F7-968B3CBC9BD1

Device         Start     End Sectors  Size Type
/dev/loop0p1  262144 4192255 3930112  1.9G Linux root (x86-64)
/dev/loop0p14   2048    8191    6144    3M BIOS boot
/dev/loop0p15   8192  262143  253952  124M EFI System

Partition table entries are not in disk order.
# ls -l /dev/mapper/
total 0
crw------- 1 root root 10, 236 Apr  2 22:45 control
lrwxrwxrwx 1 root root       7 Apr  2 23:19 loop0p1 -> ../dm-0
lrwxrwxrwx 1 root root       7 Apr  2 23:19 loop0p14 -> ../dm-1
lrwxrwxrwx 1 root root       7 Apr  2 23:19 loop0p15 -> ../dm-2
# mkdir -p /mnt/loop0p1
# mkdir -p /mnt/loop0p15
# mount -t auto /dev/mapper/loop0p1 /mnt/loop0p1
# mount -t auto /dev/mapper/loop0p15 /mnt/loop0p15
# mount |grep loop
/dev/loop0p1 on /mnt/loop0p1 type ext4 (rw,relatime)
/dev/loop0p15 on /mnt/loop0p15 type vfat (rw,relatime,fmask=0002,dmask=0002,allow_utime ↩
    =0020,codepage=437,iocharset=ascii,shortname=mixed,utf8,errors=remount-ro)
...hack...hack...hack
# umount /dev/mapper/loop0p1
# umount /dev/mapper/loop0p15
# kpartx -d disk.img
```

9.7.4 Limpar um ficheiro de imagem de disco

Um ficheiro de imagem de disco, ”`disk.img`” pode ser limpo de todos os ficheiros removidos numa imagem limpa ”`new.img`” com o seguinte.

```
# mkdir old; mkdir new
# mount -t auto -o loop disk.img old
# dd bs=1 count=0 if=/dev/zero of=new.img seek=5G
# mount -t auto -o loop new.img new
# cd old
# cp -a --sparse=always ./ ../new/
```

```
# cd ..
# umount new.img
# umount disk.img
```

Se o ”disk.img” está em ext2, ext3 ou ext4, você também pode usar o zerofree(8) do pacote zerofree como a seguir.

```
# losetup --show -f disk.img
/dev/loop0
# zerofree /dev/loop0
# cp --sparse=always disk.img new.img
# losetup -d /dev/loop0
```

9.7.5 Criar um ficheiro de imagem de disco vazio

A imagem de disco vazia ”disk.img” que pode crescer até aos 5GiB pode ser feita a usar o dd(1) como a seguir.

```
$ dd bs=1 count=0 if=/dev/zero of=disk.img seek=5G
```

Em vez de utilizar dd(1), pode ser utilizado aqui o fallocate especializado (8).

Pode criar um sistema de ficheiros ext4 nesta imagem de disco ”disk.img” a usar o dispositivo loop como a seguir.

```
# losetup --show -f disk.img
/dev/loop0
# mkfs.ext4 /dev/loop0
...hack...hack...hack
# losetup -d /dev/loop0
$ du  --apparent-size -h disk.img
5.0G  disk.img
$ du -h disk.img
83M disk.img
```

Para ”disk.img”, o tamanho de ficheiro dele é 5.0 Gb e a utilização real do disco dele é apenas 83 Mb. Esta discrepância é possível porque o ext4 pode manter o ficheiro sparse.

Dica
A utilização de disco real do ficheiro sparse cresce com os dados que são escritos nele.

A usar uma operação semelhante em dispositivos criados pelo dispositivo loop ou o mapeador de dispositivos como Secção 9.7.3, pode particionar esta imagem de disco ”disk.img” a usar o parted(8) ou o fdisk(8) e pode criar um sistema de ficheiros nela a usar mkfs.ext4(8), mkswap(8), etc.

9.7.6 Criar o ficheiro de imagem ISO9660

Dica
Ambos genisoimage(1) fornecido por cdrkit e xorrisofs(1) fornecido por Libburnia partilham a mesma sintaxe de comando excepto o nome do comando.

O ficheiro de imagem ISO9660, ”cd.iso”, a partir da árvore de diretórios fonte em ”diretório_fonte” pode ser feito a usar o genisoimage(1) disponibilizado pelo cdrkit com o seguinte.

```
#  genisoimage -r -J -T -V volume_id -o cd.iso source_directory
```

De modo semelhante, o ficheiro de imagem ISO9660 de arranque, "`cdboot.iso`", pode ser feito a partir do `instalador-de` como árvore de diretórios em "`diretório_fonte`" com o seguinte.

```
#  genisoimage -r -o cdboot.iso -V volume_id \
   -b isolinux/isolinux.bin -c isolinux/boot.cat \
   -no-emul-boot -boot-load-size 4 -boot-info-table source_directory
```

Aqui é usado para arranque o boot loader Isolinux (veja Secção 3.1.2).

Pode calcular o valor md5sum e fazer a imagem ISO9660 directamente a partir do dispositivo CD-ROM como a seguir.

```
$ isoinfo -d -i /dev/cdrom
CD-ROM is in ISO 9660 format
...
Logical block size is: 2048
Volume size is: 23150592
...
# dd if=/dev/cdrom bs=2048 count=23150592 conv=notrunc,noerror | md5sum
# dd if=/dev/cdrom bs=2048 count=23150592 conv=notrunc,noerror > cd.iso
```

Atenção
Tem de ter o cuidado de evitar o bug de leitura antecipada do sistema de ficheiros ISO9660 do Linux como em cima para obter o resultado correcto.

9.7.7 Escrever directamente ao CD/DVD-R/RW

Dica
Um DVD é apenas um CD grande para o wodim(1) fornecido por cdrkit e xorrecord(1) fornecido por Libburnia. Estes comandos partilham a mesma sintaxe de comando excepto o nome do comando.

Pode procurar um dispositivo utilizável com o seguinte.

```
# wodim --devices
```

Então o CD-R vazio é inserido na drive de CD e o ficheiro de imagem ISO9660, "`cd.iso`" é escrito neste dispositivo, ex. "`/dev/sda`", a usar o wodim(1) com o seguinte.

```
# wodim -v -eject dev=/dev/sda cd.iso
```

Se for usado um CD-RW em vez de um CD-R, faça antes o seguinte.

```
# wodim -v -eject blank=fast dev=/dev/sda cd.iso
```

Dica
Se o seu ambiente montar CDs automaticamente, desmonte-o com "`sudo umount /dev/sda`" a partir da consola antes de usar o wodim(1).

9.7.8 Montar o ficheiro de imagem ISO9660

Se "`cd.iso`" conter uma imagem ISO9660, então o seguinte monta-o manualmente em "`/cdrom`".

```
# mount -t iso9660 -o ro,loop cd.iso /cdrom
```

Dica
Os sistemas de ambiente de trabalho modernos podem montar medias amovíveis, tais como um CD formatado em ISO9660, automaticamente (veja Secção 10.1.7).

9.8 Os dados binários

Aqui, discutimos manipulação directa de dados binários em meios de armazenamento.

9.8.1 Ver e editar dados binários

o método de visualização mais básico de dados binários é usar o comando "`od -t x1`".

pacote	popcon	tamanho	descrição
`coreutils`	V:912, I:1000	17994	pacote básico que tem od(1) para despejar ficheiros (HEX, ASCII, OCTAL, …)
`bsdmainutils`	V:4, I:121	17	pacote utilitário que tem hd(1) para despejar ficheiros (HEX, ASCII, OCTAL, …)
`hexedit`	V:0.8, I:7.5	70	editor binário e visualizador (HEX, ASCII)
`bless`	V:0.1, I:1.2	924	editor hexadecimal cheiro de funcionalidades (GNOME)
`okteta`	V:1, I:11	1591	editor hexadecimal cheiro de funcionalidades (KDE4)
`ncurses-hexedit`	V:0.1, I:1.1	130	editor binário e visualizador (HEX, ASCII, EBCDIC)
`beav`	V:0.04, I:0.26	137	editor binário e visualizador (HEX, ASCII, EBCDIC, OCTAL, …)

Tabela 9.21: Lista de pacote para ver e editar dados binários

Dica
HEX é usado como um acrónimo para o formato hexadecimal com radix 16. OCTAL é para formato octal com radix 8. ASCII é para American Standard Code for Information Interchange, isto é, código de texto Inglês normal. EBCDIC é para Extended Binary Coded Decimal Interchange Code usado em sistemas operativos com infraestrutura da IBM.

9.8.2 Manipular ficheiros sem montar o disco

Existem ferramentas para ler e escrever ficheiros sem montar o disco.

9.8.3 Redundância de dados

Os sistemas RAID por software oferecidos pelo kernel Linux oferecem redundância de dados ao nível do sistema de ficheiros do kernel, para se conseguir altos níveis de fiabilidade de armazenamento.

Também existem ferramentas para adicionar redundância a ficheiros ao nível de programa aplicação, para se conseguir altos níveis de fiabilidade de armazenamento.

pacote	popcon	tamanho	descrição
mtools	V:7, I:52	400	utilitários para sistemas de ficheiros MSDOS sem os montar
hfsutils	V:0.1, I:2.5	178	utilitários para sistemas de ficheiros HFS e HFS+ sem os montar

Tabela 9.22: Lista de pacotes para ler e escrever ficheiros sem montar o disco

pacote	popcon	tamanho	descrição
par2	V:9, I:116	330	Parity Archive Volume Set, para verificação e reparação de ficheiros
dvdisaster	V:0.1, I:1.1	1422	protecção de dados contra percas/riscos/envelhecimento para medias CD/DVD
dvbackup	V:0.01, I:0.03	413	ferramenta de backup que usa câmaras de filmar MiniDV (a disponibilizar rsbep(1))

Tabela 9.23: Lista de ferramentas para adicionar redundância de dados a ficheiros

9.8.4 Recuperação de ficheiros e dados e análise forense

Existem ferramentas para recuperação de ficheiros e dados e análise forense.

pacote	popcon	tamanho	descrição
testdisk	V:2, I:25	528	utilitários para sondagem de partições e recuperação de discos
magicrescue	V:0.2, I:1.7	257	utilitário para recuperar ficheiros ao procurar por bytes mágicos
scalpel	V:0.2, I:2.2	89	frugal, entalhador de ficheiros de alta performance
myrescue	V:0.2, I:1.9	81	recuperar dados de discos rígidos danificados
extundelete	V:0.5, I:7.3	152	utilitários para recuperar ficheiros apagados no sistema de ficheiros ext3/4
ext4magic	V:0.2, I:3.5	235	utilitários para recuperar ficheiros apagados no sistema de ficheiros ext3/4
ext3grep	V:0.2, I:1.8	299	ferramenta para ajudar a recuperar ficheiros apagados no sistema de ficheiros ext3
scrounge-ntfs	V:0.1, I:1.6	49	programa de recuperação de dados para sistemas de ficheiros NTFS
gzrt	V:0.02, I:0.43	34	conjunto de ferramentas de recuperação gzip
sleuthkit	V:3, I:25	1119	ferramentas para análise forense. (Sleuthkit)
autopsy	V:0.1, I:1.1	1026	interface gráfica para o SleuthKit
foremost	V:0.3, I:3.8	101	aplicação forense para recuperar dados
guymager	V:0.08, I:0.71	1049	ferramenta de imagem forense baseada em Qt
dcfldd	V:0.3, I:2.6	113	versão melhorada do dd para forenses e segurança

Tabela 9.24: Lista de pacotes para recuperação de ficheiros e dados e análise forense

Dica
Pode recuperar ficheiros apagados no sistema de ficheiros ext2 a usar os comandos `list_deleted_inodes` e `undel` de debugfs(8) no pacote `e2fsprogs`.

9.8.5 Dividir um ficheiro grande em ficheiros pequenos

Quando os dados são muito grandes para fazer backup num ficheiro único, pode fazer backup ao conteúdo dele após dividi-lo em fatias de, por exemplo, 2000Mb e mais tarde fundir essas fatias de volta para o ficheiro original.

```
$ split -b 2000m large_file
$ cat x* >large_file
```



Cuidado
Por favor certifique-se que não tem nenhuns ficheiros que começam com ”x” para evitar falhas com nomes.

9.8.6 Limpar conteúdo de ficheiro

De modo a limpar o conteúdo de um ficheiro como um ficheiro log, não use o rm(1) para apagar o ficheiro e depois crie um ficheiro vazio, porque o ficheiro pode ainda estar a ser acedido no intervalo entre comandos. O seguinte é o modo seguro de limpar o conteúdo do ficheiro.

```
$ :>file_to_be_cleared
```

9.8.7 Ficheiros dummy

Os seguintes comandos criam ficheiros dummy ou vazios.

```
$ dd if=/dev/zero    of=5kb.file bs=1k count=5
$ dd if=/dev/urandom of=7mb.file bs=1M count=7
$ touch zero.file
$ : > alwayszero.file
```

Deve encontrar os seguintes ficheiros.

- ”`5kb.file`” é 5KB de zeros.
- ”`7mb.file`” são 7MB de dados aleatórios.
- ”`zero.file`” pode ser um ficheiro de 0 bytes. Se existir, o `mtime` dele é atualizado enquanto o conteúdo e tamanho dele são mantidos.
- ”`alwayszero.file`” é sempre um ficheiro de 0 bytes. Se existir, o `mtime` dele é atualizado e o conteúdo dele é reposto.

9.8.8 apagar um disco rígido inteiro

Existem várias maneiras de apagar completamente os dados dum dispositivo inteiro tipo disco rígido, ex., Memória flash USB em ”`/dev/sda`”.



Cuidado
Primeiro verifique a localização da sua Memória flash USB com o mount(8) antes de executar os comandos aqui. O dispositivo apontado por ”`/dev/sda`” pode ser um disco rígido SCSI ou SATA onde pode residir todo o seu sistema.

Apagar todo o conteúdo do disco ao repor os dados a 0 com o seguinte.

```
# dd if=/dev/zero of=/dev/sda
```

Apagar tudo ao sobrescrever com dados aleatórios como a seguir.

```
# dd if=/dev/urandom of=/dev/sda
```

Apagar tudo muito eficientemente ao sobrescrever com dados aleatórios como a seguir.

```
# shred -v -n 1 /dev/sda
```

Em alternativa, pode utilizar badblocks(8) com a opção `-t random`.

Como o dd(1) está disponível a partir da shell de muitos CDs de arranque de Linux como o CD de instalação de Debian, pode apagar completamente o seu sistema instalado no disco rígido, por exemplo, "`/dev/sda`", "`/dev/sda`", etc., ao correr um comando de limpeza a partir de tal media de arranque.

9.8.9 Apagar uma área não utilizada do disco rígido

A área não utilizada de um disco rígido (ou Memória flash USB), por exemplo "`/dev/sdb1`" pode ainda conter os próprios dados apagados pois eles são apenas 'desligados' do sistema de ficheiros. Estes podem ser limpos ao sobrescrever a área onde estão.

```
# mount -t auto /dev/sdb1 /mnt/foo
# cd /mnt/foo
# dd if=/dev/zero of=junk
dd: writing to `junk': No space left on device
...
# sync
# umount /dev/sdb1
```



Atenção
Normalmente isto é suficientemente bom para a sua Memória flash USB. Mas não é perfeito. A maioria das partes dos nomes de ficheiros apagados e os atributos deles podem ficar escondidos e permanecerem no sistema de ficheiros.

9.8.10 Recuperar ficheiros apagados mas ainda abertos

Mesmo que tenha acidentalmente apagado um ficheiro, desde que esse ficheiro esteja ainda a ser usado por alguma aplicação (em modo de leitura ou escrita), é possível recuperar tal ficheiro.

Por exemplo, tente o seguinte:

```
$ echo foo > bar
$ less bar
$ ps aux | grep ' less[ ]'
bozo    4775  0.0  0.0  92200   884 pts/8    S+   00:18   0:00 less bar
$ rm bar
$ ls -l /proc/4775/fd | grep bar
lr-x------ 1 bozo bozo 64 2008-05-09 00:19 4 -> /home/bozo/bar (deleted)
$ cat /proc/4775/fd/4 >bar
$ ls -l
-rw-r--r-- 1 bozo bozo 4 2008-05-09 00:25 bar
$ cat bar
foo
```

Execute em outro terminal (quando tem o pacote `lsof` instalado) o seguinte.

```
$ ls -li bar
2228329 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:02 bar
$ lsof |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar
$ rm bar
$ lsof |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar (deleted)
$ cat /proc/4775/fd/4 >bar
$ ls -li bar
2228302 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:05 bar
$ cat bar
foo
```

9.8.11 Procurar todas as ligações rígidas

Os ficheiros com ligações rígidas podem ser identificados com "`ls -li`".

```
$ ls -li
total 0
2738405 -rw-r--r-- 1 root root 0 2008-09-15 20:21 bar
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 baz
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 foo
```

Ambos "`baz`" e "`foo`" têm contagens de ligações de "2" (>1) a mostrar que têm ligaçõesrígidas. Os números de inode deles são comuns "2738404". Isto significa que são o mesmo ficheiro em ligação rígida. Se não encontrar todos os ficheiros em ligação rígida por acaso, pode procurá-los pelo inode, ex., "2738404" com o seguinte.

```
# find /path/to/mount/point -xdev -inum 2738404
```

9.8.12 Consumo invisível do espaço do disco

Todos os ficheiros apagadas mas abertos consomem espaço no disco apesar de não estarem visíveis ao du(1) normal. Eles podem ser listados com o tamanho deles com o seguinte.

```
# lsof -s -X / |grep deleted
```

9.9 Dicas de encriptação de dados

Com acesso físico ao seu PC, qualquer um pode facilmente ganhar privilégios de root e aceder a todos os ficheiros no seu PC (veja Secção 4.6.4). Isto significa que o sistema de palavra passe no login não pode proteger os seus dados privados e sensíveis contra um possível roubo do seu PC. Tem que implementar uma tecnologia de encriptação de dados para o fazer. Apesar do GNU privacy guard (veja Secção 10.3) poder encriptar ficheiro,consome alguns esforços do utilizador.

Dm-crypt facilitam a encriptação de dados automática nativamente através de módulos do kernel Linux com o mínimo de esforço do utilizador utilizando device-mapper.



Cuidado
A encriptação de dados custa tempo de CPU, etc. Os dados encriptados ficam inacessíveis se a palavra-passe for perdida. Pondere as suas vantagens e custos.

pacote	popcon	tamanho	descrição
cryptsetup	V:34, I:77	469	utilitários para dispositivos de bloco encriptados (dm-crypt / LUKS)
cryptmount	V:1.8, I:2.3	236	utilitários para dispositivos de bloco encriptados (dm-crypt / LUKS) com focagem na montagem/desmontagem por utilizadores normais
fscrypt	V:0.4, I:1.1	6844	utilitários para a encriptação do sistema de ficheiros Linux (fscrypt)
libpam-fscrypt	V:0.31, I:0.34	6941	Módulo PAM para encriptação do sistema de ficheiros Linux(fscrypt)

Tabela 9.25: Lista de utilitários de encriptação de dados

Nota
O sistema Debian inteiro pode ser instalado num disco encriptado pelo instalador debian (lenny ou mais recente) a usar dm-crypt/LUKS e initramfs.

Dica
Veja Secção 10.3 para utilitário de encriptação do espaço de utilizador: GNU Privacy Guard.

9.9.1 Encriptação de discos amovíveis com dm-crypt/LUKS

Pode encriptar o conteúdo de dispositivos de massa amovíveis, por exemplo, uma Memória flash USB em "`/dev/sdx`", a usar dm-crypt/LUKS. Simplesmente formate-a como se segue.

```
# fdisk /dev/sdx
... "n" "p" "1" "return" "return" "w"
# cryptsetup luksFormat /dev/sdx1
...
# cryptsetup open /dev/sdx1 secret
...
# ls -l /dev/mapper/
total 0
crw-rw---- 1 root root  10, 60 2021-10-04 18:44 control
lrwxrwxrwx 1 root root       7 2021-10-04 23:55 secret -> ../dm-0
# mkfs.vfat /dev/mapper/secret
...
# cryptsetup close secret
```

Depois, pode ser montado normalmente em "`/media/`*`username/disk_label`*", exceto para pedir a password (veja Secção 10.1.7) no ambiente de trabalho moderno utilizando o pacote `udisks2`. A diferença é que todos os dados escritos nele são encriptados. A introdução da palavra-passe pode ser automatizada utilizando um porta-chaves (ver Secção 10.3.6).

Pode, em alternativa, formatar os media num sistema de ficheiros diferente, por exemplo, ext4 com "`mkfs.ext4 /dev/mapper/sdx1`". Se em vez disso for utilizado o btrfs, o pacote `udisks2-btrfs` precisa de ser instalado. Para estes sistemas de ficheiros, a propriedade do ficheiro e as permissões podem ter de ser configuradas.

9.9.2 Montar discos encriptados com dm-crypt/LUKS

Por exemplo, uma partição de disco criada com dm-crypt/LUKS em "`/dev/sdc5`" pelo instalador Debian pode ser montada em `/mnt`" como a seguir:

```
$ sudo cryptsetup open /dev/sdc5 ninja --type luks
Enter passphrase for /dev/sdc5: ****
$ sudo lvm
lvm> lvscan
  inactive          '/dev/ninja-vg/root' [13.52 GiB] inherit
  inactive          '/dev/ninja-vg/swap_1' [640.00 MiB] inherit
  ACTIVE            '/dev/goofy/root' [180.00 GiB] inherit
  ACTIVE            '/dev/goofy/swap' [9.70 GiB] inherit
lvm> lvchange -a y /dev/ninja-vg/root
lvm> exit
  Exiting.
$ sudo mount /dev/ninja-vg/root /mnt
```

9.10 O kernel

Debian distribui o kernel Linux organizado em módulos como pacotes para as arquitecturas suportadas.

Se está a ler esta documentação, provavelmente não precisa de compilar o kernel Linux por si próprio.

9.10.1 Parâmetros do kernel

Muitas funcionalidades do Linux são configuráveis via parâmetros de kernel como a seguir.

- Parâmetros de kernel iniciados pelo gestor de arranque (veja Secção 3.1.2)
- Parâmetros de kernel alterados pelo sysctl(8) durante a execução para os acessíveis via sysfs (veja Secção 1.2.12)
- Parâmetros de módulos definidos por argumentos do modprobe(8) quando um módulo é activado (veja Secção 9.7.3)

Ver "Guia do utilizador e do administrador do kernel Linux » Os parâmetros da linha de comandos do kernel" para mais detalhes.

9.10.2 Cabeçalhos do kernel

A maioria dos **programas normais** não precisa dos cabeçalhos do kernel e na verdade podem bloquear se os usar directamente para compilação. Eles devem ser compilados contra os cabeçalhos em "`/usr/include/linux`" e "`/usr/include/asm`" disponibilizado pelo pacote `libc6-dev` (criado a partir do pacote fonte `glibc`) no sistema Debian.

Nota
Para compilar alguns programas específicos do kernel, tais como os módulos do kernel a partir da fonte externa e o daemon automounter (`amd`), tem de incluir o caminho para os cabeçalhos correspondentes do kernel, e.g. "`-I/usr/src/linux-particular-version/include/`", na sua linha de comandos.

9.10.3 Compilar o kernel e módulos relacionados

O Debian tem método próprio dele para compilar o kernel e os módulos relacionados.

Se usa `initrd` em Secção 3.1.2, certifique-se de ler a informação relacionada em initramfs-tools(8), update-initramfs(8), mkinitramfs(8) e initramfs.conf(5).

pacote	popcon	tamanho	descrição
build-essential	V:20, I:510	17	pacotes essenciais para construir pacotes Debian: make, gcc, ...
bzip2	V:163, I:973	113	utilitários de compressão e descompressão para ficheiros bz2
libncurses5-dev	I:31	6	bibliotecas de programadores e documentos para ncurses
git	V:422, I:625	52512	git: sistema de controle de versão distribuído usado pelo kernel Linux
fakeroot	V:32, I:511	225	disponibiliza um ambiente de falso-root para construção de pacotes como não-root
initramfs-tools	V:495, I:989	49	ferramenta para construir uma initramfs (específico de Debian)
dkms	V:73, I:147	235	suporte de módulos de kernel dinâmicos (DKMS) (genérico)
module-assistant	V:1, I:12	395	ferramenta auxiliar para fazer pacotes de módulos (específico para Debian)
devscripts	V:5, I:31	2783	scripts de ajuda para um responsável de pacote Debian (específico de Debian)

Tabela 9.26: Lista de pacotes chave a serem instalados para a recompilação do kernel no sistema Debian



Atenção
Não ponha ligações simbólicas aos diretórios na árvore fonte (ex. "/usr/src/linux*") a partir de "/usr/include/linux" e "/usr/include/asm" quando compilar a fonte do kernel Linux. (Alguns documentos antigos sugerem isto.)

Nota
Quando compilar o kernel Linux mais recente no sistema Debian stable, pode ser necessário o uso das ferramentas backport mais recentes do Debian unstable.
module-assistant(8) (ou a sua forma curta m-a) ajuda os utilizadores a construir e instalar facilmente pacote(s) de módulos para um ou mais kernels personalizados.
O suporte dinâmico a módulos do kernel (DKMS) é uma nova infraestrutura independente da distribuição desenhada para permitir que módulos de kernel individuais sejam atualizados sem se alterar todo o kernel. Isto é usado para a manutenção de módulos de fora-da-árvore. Isto também facilita a reconstrução de módulos quando se atualiza os kernels.

9.10.4 Compilar código-fonte do kernel: a recomendação da equipa do kernel de Debian

Para compilar pacotes binários de kernels personalizados a partir do código-fonte original, deve utilizar o alvo disponibilizado por "deb-pkg".

```
$ sudo apt-get build-dep linux
$ cd /usr/src
$ wget https://mirrors.edge.kernel.org/pub/linux/kernel/v7.x/linux-7.2.8.tar.xz
$ tar --xz -xvf linux-7.2.8.tar.xz
$ cd linux-7.2.8
$ cp /boot/config-7.2.8.config
$ make menuconfig
 ...
$ make deb-pkg
```

Dica
O pacote linux-source-*version* disponibiliza o código-fonte do kernel Linux com os patches Debian como "`/usr/src/linux-`*`version`*`.tar.bz2`".

Para construir pacotes binários específicos a partir do pacote de código-fonte do kernel Debian, deve utilizar os alvos "`binary-arch_`*`architecture_featureset_flavour`*"em "`debian/rules.gen`".

```
$ sudo apt-get build-dep linux
$ apt-get source linux
$ cd linux-7.*
$ fakeroot make -f debian/rules.gen binary-arch_i386_none_686
```

Veja mais informação:

- Wiki Debian: KernelFAQ
- Wiki Debian: DebianKernel
- Manual Debian do Kernel Linux: https://kernel-handbook.debian.net

9.10.5 Controladores de hardware e firmware

O driver de hardware é o código que corre nos CPUs principais do sistema alvo. A maioria dos drivers de hardware estão agora disponíveis como software livre e estão incluídos nos pacotes normais do kernel Debian na área `main`.

- Controlador de GPU
  - Controlador de GPU Intel (`main`)
  - Controlador de GPU AMD/ATI (`main`)
  - Controlador de GPU NVIDIA (`main` nouveau e em `non-free` controladores binários, sem código fonte, suportados pelo fabricante.)

O firmware é o código ou os dados carregados no dispositivo ligado ao sistema de destino (por exemplo, micro-código da CPU, código de renderização executado na GPU, ou dados FPGA / CPLD, ...). Alguns pacotes de firmware estão disponíveis como software livre, mas muitos pacotes de firmware não estão disponíveis como software livre, uma vez que contêm dados binários sem fonte. A instalação destes dados de firmware é essencial para que o dispositivo funcione como esperado.

- Os pacotes de dados do firmware que contêm dados carregados na memória volátil do dispositivo de destino.
  - `firmware-linux-free` (`main`)
  - `firmware-linux-nonfree` (`non-free-firmware`)
  - `firmware-linux-*` (`non-free-firmware`)
  - `*-firmware` (`non-free-firmware`)
  - `intel-microcode` (`non-free-firmware`)
  - `amd64-microcode` (`non-free-firmware`)
- Os pacotes de programas de atualização do firmware que actualizam os dados na memória não volátil do dispositivo alvo.
  - fwupd (`principal`): Daemon de atualização de firmware que descarrega dados de firmware do Linux Vendor Firmware Service.
  - gnome-firmware (`main`): Interface GTK para o fwupd
  - plasma-discover-backend-fwupd (`main`): Interface Qt para fwupd

Por favor note que o acesso aos pacotes `non-free-firmware` são disponibilizados pelo meio de instalação oficial para oferecer uma experiência de instalação funcional ao utilizador desde Debian 12 Bookworm. A área `non-free-firmwa` é descrita em Secção 2.1.5.

Tenha também em atenção que os dados de firmware descarregados pelo fwupd a partir do Linux Vendor Firmware Service e carregados para o kernel Linux em execução podem ser `não-livres`.

9.11 Sistema virtualizado

O uso de sistema virtualizado permite-nos correr várias instâncias do sistema simultâneamente num único hardware.

Dica
Veja o wiki Debian sobre SystemVirtualization.

9.11.1 Ferramentas de virtualização e emulação

Existem várias plataformas de ferramentas de virtualização e emulação.

- Pacotes completos de emulação de hardware, como os instalados pelo meta-pacote games-emulator
- Principalmente emulação de nível de CPU com algumas emulações de dispositivos de E/S, como o QEMU
- Principalmente virtualização no nível da CPU com algumas emulações de dispositivos de E/S, como a Máquina Virtual baseada em Kernel (KVM)
- Virtualização de contentores ao nível do SO com suporte ao nível do kernel, como LXC (Linux Containers), Docker, systemd-nspawn(1), ...
- Virtualização do acesso ao sistema de ficheiros ao nível do SO com a substituição da chamada da biblioteca do sistema no caminho do ficheiro, como o chroot
- Virtualização do acesso ao sistema de ficheiros ao nível do SO com a substituição da chamada da biblioteca do sistema sobre a propriedade do ficheiro, como o fakeroot
- Emulação da API do SO, como o Wine
- Virtualização ao nível do interpretador com a sua seleção de executáveis e substituições de bibliotecas em tempo de execução, tais como virtualenv e venv para Python

A virtualização de contentores utiliza Secção 4.7.5 e é a tecnologia por trás de Secção 7.7.

Aqui estão alguns pacotes para o ajudar a configurar o sistema virtualizado.

Veja o artigo da Wikipedia Comparação de plataformas de máquinas virtuais para uma comparação detalhada das diferentes soluções de plataformas de virtualização.

9.11.2 Fluxo de trabalho da virtualização

Nota
Os kernels predefinidos de Debian suportam KVM desde `lenny`.

O fluxo e trabalho típico para virtualização envolve vários passos.

pacote	popcon	tamanho	descrição
coreutils	V:912, I:1000	17994	Utilitários do núcleo GNU que contêm chroot(8)
util-linux	V:913, I:1000	4826	utilitários de sistema variados que contêm unshare(1)
systemd-container	V:74, I:77	2906	ferramentas de contentor/nspawn do systemd que contêm systemd-nspawn(1)
schroot	V:7.2, I:9.1	2222	ferramenta especializada para executar pacotes binários Debian em chroot
sbuild	V:3.9, I:6.9	155	ferramenta para construir pacotes binários Debian a partir de fontes Debian
debootstrap	V:4, I:44	332	bootstrap um sistema Debian básico (escrito em sh)
mmdebstrap	V:8, I:12	592	bootstrap um sistema Debian (escrito em Perl)
cdebootstrap	V:0.1, I:1.3	114	bootstrap um sistema Debian (escrito em C)
cloud-image-utils	V:1, I:14	52	utilitários de gestão de imagens na nuvem
cloud-guest-utils	V:5, I:23	67	utilitários para hóspedes na nuvem
virt-manager	V:12, I:48	2309	Virtual Machine Manager: aplicação de ambiente de trabalho para gerir máquinas virtuais
libvirt-clients	V:45, I:71	1168	programas para a biblioteca libvirt
docker.io	V:50, I:53	80705	docker: tempo de execução do contentor Linux
podman	V:34, I:38	85864	podman: motor para executar contentores baseados em OCI em Pods
podman-docker	V:2.8, I:3.4	271	motor para executar contentores baseados em OCI em Pods - revestimento para docker
incus	V:0.8, I:3.4	24	Incus: contentor de sistema e gestor de máquinas virtuais
games-emulator	I:0.19	21	emulador de jogos: Emuladores Debian para jogos
bochs	V:0.04, I:0.59	8469	Bochs: emulador PC IA-32
qemu-system	I:20	69	QEMU: binários de emulação de sistema completo
qemu-user	V:5.6, I:9.3	472213	QEMU: binários de emulação em modo de utilizador
qemu-utils	V:14, I:114	12643	QEMU: utilitários
qemu-system-x86	V:52, I:98	69257	KVM: virtualização completa em hardware x86 com virtualização assistida por hardware
virtualbox	V:2.9, I:3.7	151079	VirtualBox: solução de virtualização x86 em i386 e amd64
gnome-boxes	V:1.2, I:6.1	7074	Boxes: Aplicação GNOME simples para aceder a sistemas virtuais
xen-tools	V:0.1, I:1.4	719	ferramentas para gerir o servidor virtual XEN do debian
wine	V:13, I:54	204	Wine: Windows API Implementation (suite standard)
dosbox	V:1, I:11	2697	DOSBox: emulador x86 com gráficos Tandy/Herc/CGA/EGA/VGA/SVGA, som e DOS
lxc	V:10, I:13	1681	Ferramentas de espaço do utilizador de contentores Linux
python3-venv	V:8, I:159	6	venv para criar ambientes virtuais python (biblioteca de sistema)
python3-virtualenv	V:7, I:37	374	virtualenv para criar ambientes python virtuais isolados
pipx	V:8, I:50	4412	pipx para instalar aplicações python em ambientes isolados

Tabela 9.27: Lista de ferramentas de virtualização

- Criar um sistema de ficheiros vazio (uma árvore de ficheiros ou uma imagem de disco).
  - A árvore de ficheiros pode ser criada por "`mkdir -p /path/to/chroot`".
  - A imagem de disco crua pode ser criada com o dd(1) (veja Secção 9.7.1 e Secção 9.7.5).
  - qemu-img(1) pode ser usado para criar e converter ficheiros de imagem de disco suportados pelo QEMU.
  - Os formatos de ficheiro cru e VMDK podem ser usados como formatos comuns entre ferramentas de virtualização.
- Montar a imagem de disco com mount(8) no sistema de ficheiros (opcional).
  - Para o ficheiro de imagem de disco cru, monte-o como dispositivo loop ou dispositivo do device mapper (veja Secção 9.7.3).
  - Para imagens de disco suportadas pelo QEMU, monte-as como dispositivos de bloco de rede (veja Secção 9.11.3).
- Povoar o sistema de ficheiros alvo com os dados de sistema necessários.
  - O uso de programas como o `debootstrap` e o `cdebootstrap` ajudam com este processo (veja Secção 9.11.4).
  - Use instaladores de SOs sob o emulador de sistema completo.
- Correr um programa sob um ambiente virtualizado.
  - chroot disponibiliza um ambiente virtualizado básico suficiente para compilar programas, correr aplicações de consola e correr daemons nele.
  - QEMU oferece emulação de CPU de várias plataformas.
  - QEMU com KVM oferece emulação de sistema completo pela virtualização assistida a hardware.
  - VirtualBox oferece emulação de sistema completo em i386 ou amd64 com ou sem a virtualização assistida a hardware.

9.11.3 Montar o ficheiro de imagem de disco virtual

Para o ficheiro de imagem de disco raw, veja Secção 9.7.

Para outros ficheiros de imagem de disco virtual, pode usar o `qemu-nbd` para exportá-los pelo protocolo dispositivo de bloco de rede e montá-los a usar o módulo de kernel `nbd`.

O qemu-nbd(8) suporta os formatos de disco suportados pelo QEMU: raw, qcow2, qcow, vmdk, vdi, bochs, cow (modo-de-utilizador de Linux de copiar-ao-escrever), parallels, dmg, cloop, vpc, vvfat (VFAT virtual) e dispositivo_máquina.

O dispositivo de bloco em rede pode suportar partições do mesmo modo que o dispositivo de loop (veja Secção 9.7.3). Pode montar a primeira partição de "`disk.img`" como a seguir.

```
# modprobe nbd max_part=16
# qemu-nbd -v -c /dev/nbd0 disk.img
...
# mkdir /mnt/part1
# mount /dev/nbd0p1 /mnt/part1
```

Dica
Pode exportar apenas a primeira partição de "`disk.img`" a usar a opção "`-P 1`" para qemu-nbd(8).

9.11.4 Sistema chroot

Se deseja experimentar um novo ambiente Debian a partir de uma consola terminal, eu recomendo que use chroot. Isto permite-lhe correr aplicações de consola em Debian `unstable` e `testing` sem os riscos habituais associados e sem reiniciar. chroot(8) é a forma mais básica.



Cuidado
Os exemplos abaixo assumem que tanto o sistema base como o sistema chroot partilham a mesma arquitetura de CPU `amd64`.

Embora possa criar manualmente um ambiente chroot(8) utilizando o debootstrap(1), isto requer esforços não triviais.

O pacote sbuild para construir pacotes Debian a partir da fonte usa o ambiente chroot gerido pelo pacote schroot. Vem com o script auxiliar sbuild-createchroot(1). Vamos aprender como ele funciona, executando-o da seguinte forma.

```
$ sudo mkdir -p /srv/chroot
$ sudo sbuild-createchroot -v --include=eatmydata,ccache unstable /srv/chroot/unstable- ↩
    amd64-sbuild http://deb.debian.org/debian
 ...
```

Pode ver como debootstrap(8) preenche os dados do sistema para o ambiente `unstable` em "`/srv/chroot/unstable-a` para um sistema de compilação mínimo.

Pode iniciar sessão neste ambiente utilizando schroot(1).

```
$ sudo schroot -v -c chroot:unstable-amd64-sbuild
```

Veja como é criada uma shell de sistema a correr num ambiente `unstable`.

Nota
O ficheiro "`/usr/sbin/policy-rc.d`" que sai sempre com 101 impede que os programas daemon sejam iniciados automaticamente no sistema Debian. Veja "`/usr/share/doc/init-system-helpers/README.policy-rc.d.gz`".

Nota
Alguns programas sob chroot podem requerer acesso a mais ficheiros do sistema pai para funcionarem do que o `sbuild-createchroot` disponibiliza como acima. Por exemplo, "`/sys`", "`/etc/passwd`", "`/etc/group`", "`/var/run/utmp`", "`/var/log/wtmp`", etc. podem precisar de ser montados em união ou copiados.

Dica
O pacote `sbuild` ajuda a construir um sistema chroot e constrói um pacote dentro do chroot usando `schroot` como seu backend. É um sistema ideal para verificar dependências de construção. Veja mais sobre o sbuild na wiki Debian e um exemplo de configuração do sbuild em "Guia para administradores Debian".

Dica
O comando systemd-nspawn(1) ajuda a executar um comando ou SO num contentor leve de forma semelhante ao `chroot`. Ele é mais poderoso, pois usa espaços de nomes para virtualizar completamente a árvore de processos, IPC, nome de máquina, nome de domínio e, opcionalmente, rede e bases de dados de utilizadores. Veja systemd-nspawn.

9.11.5 Sistemas de vários ambientes de trabalho

Se deseja experimentar um novo ambiente de trabalho GUI de qualquer SO, eu recomendo que use QEMU ou KVM num sistema Debian `stable` para correr múltiplos sistemas de desktop de forma segura usando virtualização. Estes permitem-lhe correr quaisquer aplicações de área de trabalho incluindo as da Debian `unstable` e `testing` sem os riscos habituais associados a elas e sem reiniciar.

Como o QEMU puro é muito lento, é recomendado acelerá-lo com KVM quando o sistema da máquina o suporta.

Virtual Machine Manager, também conhecido como `virt-manager`, é uma ferramenta GUI conveniente para gerir máquinas virtuais KVM através da libvirt.

A imagem de disco virtual "`virtdisk.qcow2`" que contem um sistema Debian para o QEMU pode ser criada a usar o instalador de debian em pequenos CDs como a seguir.

```
$ wget https://cdimage.debian.org/debian-cd/5.0.3/amd64/iso-cd/debian-503-amd64-netinst.iso
$ qemu-img create -f qcow2 virtdisk.qcow2 5G
$ qemu -hda virtdisk.qcow2 -cdrom debian-503-amd64-netinst.iso -boot d -m 256
...
```

Dica
Correr outras distribuições de GNU/Linux como o Ubuntu e o Fedora sob virtualização é um bom modo de aprender dicas de configuração. Também outros SOs proprietários podem correr muito bem sob esta virtualização do GNU/Linux.

Veja mais dicas em Debian wiki: SystemVirtualization.

Capítulo 10

Gestão de dados

São descritas ferramentas e dicas para gerir dados binários e de texto no sistema Debian.

10.1 Partilhar, copiar e arquivar



Atenção
O acesso de escrita descoordenado a dispositivos acedidos activamente e a ficheiros a partir de múltiplos processos não deve ser feito para evitar a condição de competição. Devem ser usados mecanismos de bloqueio de ficheiro que usem o flock(1) para o evitar.

A segurança dos dados e a partilha controlada dele têm vários aspectos.

- A criação de um arquivo de dados
- O acesso a armazenamento remoto
- A duplicação
- O acompanhar do histórico de modificação
- A facilitação da partilha de dados
- A prevenção de acessos não autorizados a ficheiros
- A detecção de modificação não autorizada de ficheiros

Estas podem ser realizadas a usar a combinação de algumas ferramentas.

- Ferramentas de arquivo e compressão
- Ferramentas de cópia de sincronização
- Sistemas de ficheiros de rede
- Media de armazenamento amovível
- A shell segura
- O sistema de autenticação
- Ferramentas de sistema de controle de versão
- Ferramentas de hash e encriptação criptográfica

10.1.1 Ferramentas de arquivo e compressão

Aqui está um sumário das ferramentas de arquivo e compressão disponíveis no sistema Debian.



Atenção
Não defina a variável "$TAPE" a menos que saiba com o que esperar. Altera o comportamento do tar(1).

- O arquivo tar(1) gzipado usa a extensão de ficheiro "`.tgz`" ou "`.tar.gz`".
- O arquivo tar(1) comprimido em xz usa a extensão de ficheiro "`.txz`" ou "`.tar.xz`".
- Método de compressão popular em ferramentas FOSS tal como o tar(1) têm se movido como a seguir: `gzip` → `bzip2` → `xz`
- cp(1), scp(1) e tar(1) podem ter algumas limitações para ficheiros especiais. cpio(1) é o mais versátil.
- O cpio(1) é desenhado para ser utilizado com o find(1) e outros comandos e apropriado para criar scripts de backup pois a parte de seleção de ficheiros do script pode ser testada independentemente.
- A estrutura interna dos ficheiros de dados do Libreoffice são ficheiros "`.jar`" que também podem ser abertos pelo `unzip`.
- A ferramenta de arquivo que é "de-facto" multi-plataforma é o `zip`. Use-o como "`zip -rX`" para obter o máximo de compatibilidade. Use também a opção "`-s`", se o tamanho máximo de ficheiro for importante.

10.1.2 Ferramentas de cópia de sincronização

Aqui está um sumário de ferramentas de cópia simples e salvaguarda disponíveis no sistema Debian.

Copiar ficheiros com o rsync(8) oferece funcionalidades mais ricas que os outros.

- algoritmo de transferência delta que envia apenas as diferenças entre os ficheiros da fonte e os ficheiros existentes no destino
- algoritmo de verificação rápida (predefinido) que procura ficheiros que alteraram no tamanho ou hora da última modificação
- opções "`--exclude`" e "`--exclude-from`" semelhantes ao tar(1)
- sintaxe de "uma barra final no diretório fonte" que evita a criação de um nível de diretório adicional no destino.

Dica
Ferramentas de sistema de controlo de versão (VCS) em Tabela 10.14 podem funcionar como a copia de multi-modos e ferramentas de sincronização.

10.1.3 Idiomas para o arquivo

Aqui estão várias maneiras de arquivar e "desarquivar" o conteúdo completo do diretório "`./source`" a usar diferentes ferramentas.

tar(1) de GNU:

```
$ tar -cvJf archive.tar.xz ./source
$ tar -xvJf archive.tar.xz
```

pacote	popcon	tamanho	extensão	comando	comentário
tar	V:907, I:1000	3101	.tar	tar(1)	o arquivador standard (de facto standard)
cpio	V:352, I:999	1201	.cpio	cpio(1)	arquivador estilo Unix System V, usar com o find(1)
binutils	V:191, I:631	1131	.ar	ar(1)	arquivador para a criação de bibliotecas estáticas
fastjar	V:1.0, I:9.5	182	.jar	fastjar(1)	arquivador para Java (estilo zip)
pax	V:5.1, I:9.0	167	.pax	pax(1)	novo arquivador standard do POSIX, um compromisso entre tar e cpio
gzip	V:895, I:1000	262	.gz	gzip(1), zcat(1), …	LZ77 utilitário de compressão do GNU (o standard de facto)
bzip2	V:163, I:973	113	.bz2	bzip2(1), bzcat(1), …	Compressão de organização de blocos de Burrows-Wheeler utilitário com um rácio de compressão mais alto que o gzip(1) (mais lento que o gzip com sintaxe semelhante)
lzma	V:0.6, I:9.6	349	.lzma	lzma(1)	LZMA utilitário de compressão com rácio de compressão mais alto que o gzip(1) (descontinuado)
xz-utils	V:310, I:981	1551	.xz	xz(1), xzdec(1), …	XZ utilitário de compressão com rácio de compressão mais alto que o bzip2(1) (mais lento que o gzip mas mais rápido que o bzip2; substituto para o utilitário de compressão LZMA)
zstd	V:334, I:830	2308	.zstd	zstd(1), zstdcat(1), …	Utilitário de compressão rápida sem perdas Zstandard
p7zip	V:5, I:172	8	.7z	7zr(1), p7zip(1)	7-Zip arquivador de arquivos com alta taxa de compressão (LZMA)
p7zip-full	V:17, I:199	12	.7z	7z(1), 7za(1)	7-Zip arquivador de ficheiros com rácio de compressão alto (LZMA compressão e outros)
lzop	V:14, I:129	164	.lzo	lzop(1)	LZO utilitário de compressão com mais alta compressão e mais rápida descompressão que o gzip(1) (rácio de compressão mais baixo que o gzip com sintaxe semelhante)
zip	V:48, I:355	635	.zip	zip(1)	InfoZIP: ferramenta de compressão e arquivo do DOS
unzip	V:123, I:744	399	.zip	unzip(1)	InfoZIP: ferramenta de descompressão e de de-arquivar do DOS

Tabela 10.1: Lista de ferramentas de arquivo e compressão

pacote	popcon	tamanho	ferramenta	função
coreutils	V:912, I:1000	17994	GNU cp	copia localmente ficheiros e diretórios (”-a” para ser recursivo)
openssh-client	V:682, I:997	3611	scp	copia remotamente ficheiros e diretórios (cliente, ”-r” para ser recursivo)
openssh-server	V:774, I:820	3615	sshd	copia ficheiros e diretórios remotamente (servidor remoto)
rsync	V:180, I:539	967		sincronização remota a salvaguarda de 1 via
unison	V:3, I:12	14		sincronização remota a salvaguarda de 2 vias

Tabela 10.2: Lista de ferramentas de cópia e sincronização

Alternativamente, pelo seguinte.

```
$ find ./source -xdev -print0 | tar -cvJf archive.tar.xz --null -T -
```

cpio(1):

```
$ find ./source -xdev -print0 | cpio -ov --null > archive.cpio; xz archive.cpio
$ zcat archive.cpio.xz | cpio -i
```

10.1.4 Idiomas para a cópia

Aqui estão algumas maneiras de copiar o conteúdo inteiro do diretório ”./source” a usar diferentes ferramentas.

- Cópia local: diretório ”./source” → diretório ”/dest”
- Cópia remota: diretório ”./source” em máquina local → diretório ”/dest” na máquina ”utilizador@máquina.domíni

rsync(8):

```
# cd ./source; rsync -aHAXSv . /dest
# cd ./source; rsync -aHAXSv . user@host.dom:/dest
```

Pode alternativamente usar a sintaxe de ”uma barra à direita no diretório fonte”.

```
# rsync -aHAXSv ./source/ /dest
# rsync -aHAXSv ./source/ user@host.dom:/dest
```

Alternativamente, pelo seguinte.

```
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . /dest
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . user@host.dom:/dest
```

cp(1) de GNU e scp(1) de openSSH:

```
# cd ./source; cp -a . /dest
# cd ./source; scp -pr . user@host.dom:/dest
```

tar(1) de GNU:

```
# (cd ./source && tar cf - . ) | (cd /dest && tar xvfp - )
# (cd ./source && tar cf - . ) | ssh user@host.dom '(cd /dest && tar xvfp - )'
```

cpio(1):

```
# cd ./source; find . -print0 | cpio -pvdm --null --sparse /dest
```

Pode substituir "`.`" por "`foo`" para todos os exemplos que contenham "`.`" para copiar ficheiros do diretório "`./source/foo`" ao diretório "`/dest/foo`".

Pode substituir "`.`" pelo caminho absoluto "`/caminho/para/fonte/foo`" para todos os exemplos que contenham "`.`" para abandonar "`cd ./source;`". Estes copiam ficheiros para localizações diferentes a depender das ferramentas utilizadas conforme a seguir.

- "`/dest/foo`": rsync(8), cp(1) do GNU e scp(1)
- "`/dest/path/to/source/foo`": GNU tar(1) e cpio(1)

Dica
rsync(8) e cp(1) do GNU têm a opção "`-u`" para saltar ficheiros que são mais recentes no receptor.

10.1.5 Idiomas para a seleção de ficheiros

O find(1) é usado para selecionar ficheiros para o arquivo e copiar comandos (veja Secção 10.1.3 e Secção 10.1.4) ou para xargs(1) (veja Secção 9.4.9). Isto pode ser melhorado ao usar os seus argumentos de comando.

A sintaxe básica de find(1) pode ser sumariada no seguinte.

- Os seus argumentos condicionais são avaliados da esquerda à direita.
- Esta avaliação pára assim que o resultado dele é determinado.
- O "**OU** lógico" (especificado por "`-o`" entre condicionais) tem precedência mais baixa que o "**E** lógico" (especificado por "`-a`" ou nada entre condicionais).
- O "**NÃO** lógico" (especificado por "`!`" antes duma condicional) tem precedência mas alta que o "**E** lógico".
- "`-prune`" retorna sempre o **VERDADEIRO** lógico e, se for um diretório, a busca de ficheiro é parada para além deste ponto.
- "`-name`" corresponde à base do nome de ficheiro com glob de shell (veja Secção 1.5.6) mas também corresponde ao "`.`" inicial dele com meta-caracteres como o "`*`" e o "?". (Nova funcionalidade do POSIX)
- "`-regex`" corresponde ao caminho completo com estilo emacs **BRE** (veja Secção 1.6.2) como predefinição.
- "`-size`" corresponde ao ficheiro baseado no tamanho do ficheiro (valor precedido de "+" para maior, precedido de "`-`" para menor)
- "`-newer`" corresponde ao ficheiro mais recente que aquele especificado no argumento dele.
- "`-print0`" retorna sempre o **TRUE** lógico e escreve o nome de ficheiro completo (terminado em nulo) na saída standard.

O find(1) é usado geralmente com um estilo idiomático como a seguir.

```
# find /path/to \
    -xdev -regextype posix-extended \
    -type f -regex ".*\.cpio|.*~" -prune -o \
    -type d -regex ".*/\.git" -prune -o \
    -type f -size +99M -prune -o \
    -type f -newer /path/to/timestamp -print0
```

Isto significa fazer as seguintes acções.

1. Procurar todos os ficheiros que começam por "`/caminho/para`"
2. Limitar globalmente a busca para dentro do sistema de ficheiros inicial e usa **ERE** (ao inves veja Secção 1.6.2)
3. Excluir da busca os ficheiros que correspondem à expressão regular "`.*\.cpio`" ou "`.*~`" ao parar o processamento
4. Excluir da busca os diretórios que correspondem à expressão regular "`.*/\.git`" ao parar o processamento
5. Exclui da busca os ficheiros maiores que 99 Mb (unidades de 1048576 bytes) ao parar o processamento
6. Escrever os nomes de ficheiros que satisfazem as condições de busca em cima e são mais recentes que "`/caminho/para/marca-temporal`"

Por favor note a utilização idiomática de "`-prune -o`" para excluir ficheiros no exemplo em cima.

Nota
Para um sistema não-Debian tipo Unix, algumas opções podem não ser suportadas pelo find(1). Em tal caso, por favor considere ajustar os métodos de correspondência e substitua "`-print0`" por "`-print`". Poderá ter que ajustar também os comandos relacionados.

10.1.6 Meio de arquivo

Quando escolher o meio de armazenamento de dados de computador para arquivar dados importantes, deverá ter cuidado com as suas limitações. Para os pequenos backups de dados pessoais uso CD-Rs e DVD-Rs de uma boa marca e guardo-os num ambiente fresco, à sombra, seco e limpo. (O meio de cassete de fita magnética parece ser popular para uso profissional.)

Nota
A segurança de resistência ao fogo destina-se a documentos de papel. A maioria dos meios de armazenamento de dados de computador têm menos tolerância à temperatura que o papel. Geralmente Confio em múltiplas cópias de segurança encriptadas em múltiplas localizações seguras.

A duração de vida optimista de meios de arquivo vista na net (a maioria é informação do fabricante).

- + de 100 anos : Papel livre de ácidos com tinta
- 100 anos : Armazenamento óptico (CD/DVD, CD/DVD-R)
- 30 anos : Armazenamento magnético (fita, disquete)
- 20 anos : Armazenamento óptico de mudança de fase (CD-RW)

Estes não contam com falhas mecânicas devido a manuseamento e etc.

Ciclos de escrita optimistas dos meios de arquivo vistos na net (a maioria é informação do fabricante).

- + de 250,000 ciclos : Disco rígido
- + de 10,000 ciclos : Memória Flash
- 1,000 ciclos : CD/DVD-RW
- 1 ciclo : CD/DVD-R, papel



Cuidado
As figuras de vida de armazenamento e ciclos de escrita mostradas aqui não devem ser usadas para decisões em qualquer armazenamento de dados crítico. Por favor consulte a informação específica do produto disponibilizada pelo fabricante.

Dica
Como os CD/DVD-R e o papel têm apenas 1 ciclo de escrita, eles previnem perdas de dados acidentais ao sobrescrever. Isto é uma vantagem!

Dica
Se precisa de um backup rápido e frequente de grandes quantidades de dados, um disco rígido numa máquina remota ligada por uma ligação de rede rápida, pode ser a única opção realista.

Dica
Se utilizar suportes regraváveis para as suas cópias de segurança, a utilização de um sistema de ficheiros como o btrfs ou zfs, que suportam instantâneos apenas de leitura, pode ser uma boa ideia.

10.1.7 Dispositivo de armazenamento amovível

Dispositivos de armazenamento amovível podem ser qualquer um dos seguintes.

- Pen USB
- Disco Rígido
- Leitor de disco óptico
- Câmara digital
- Leitor digital de música

Podem ser ligados por qualquer um dos seguintes:

- USB
- IEEE 1394 / FireWire

Os ambientes de trabalho modernos tais como o GNOME e KDE podem montar automaticamente estes dispositivos amovíveis sem uma entrada correspondente no "`/etc/fstab`".

- O pacote `udisks2` disponibiliza um daemon e utilitários associados para (des)montar esses dispositivos.
- D-bus cria eventos para iniciar processos automáticos.
- PolicyKit disponibiliza os privilégios necessários.

Dica
Os dispositivos auto-montados podem ter a opção de montagem "`uhelper=`" que é utilizada por umount(8).

Dica
A auto-montagem em ambientes de trabalho modernos apenas acontece quando esses dispositivos amovíveis não estão listados em "`/etc/fstab`".

O ponto de montagem no ambiente de trabalho moderno é escolhido como "`/media/`*`nome-utilizador/etiqueta-disc`* que pode ser personalizado da seguinte forma.

- mlabel(1) para o sistema de ficheiros FAT
- genisoimage(1) com a opção "`-V`" para o sistema de ficheiros ISO9660
- tune2fs(1) com a opção "`-L`" para sistemas de ficheiros ext2/ext3/ext4

Dica
A escolha de codificação pode necessitar de ser disponibilizada como opção de montagem (veja Secção 8.1.3).

Dica
A utilização do menu da GUI para desmontar um sistema de ficheiros pode remover o nó de dispositivo dele gerado dinamicamente tal como "`/dev/sdc`". Se desejar manter o nó de dispositivo dele, desmonte-o com o comando umount(8) na linha de comandos da shell.

10.1.8 Escolha de sistema de ficheiros para partilhar dados

Quando partilha dados com outros sistemas via dispositivos de armazenamento amovível, deve formatá-lo num sistema de ficheiros comum que seja suportado pelos dois sistemas. Aqui está uma lista de escolhas de sistemas de ficheiros.

Nota
Declarações sobre disco rijo (HDD) são aplicáveis a outros dispositivos de armazenamento tais como SSD / Pen flash USB / Cartão de memória / ... . Substitua os nomes dos dispositivos nos exemplos como `/dev/sda` com nomes de dispositivos apropriados `/dev/nvme0`, `/dev/mmcblk0`, ...

Dica
Veja Secção 9.9.1 para partilha de dados em várias plataformas a usar encriptação ao nível do dispositivo.

O sistema de ficheiros FAT é suportado pela maioria dos sistemas operativos modernos e é bastante útil para objetivos de trocas de dados via dispositivos tipo disco rígido.

Quando formatar dispositivos tipo disco rígido amovíveis para partilha de dados em multi-plataformas com o sistema de ficheiros FAT, as seguintes deverão ser escolhas seguras.

- Particioná-los com o fdisk(8), cfdisk(8) ou parted(8) (veja Secção 9.6.2) numa única partição primária e marcá-la como a seguir.
  - Tipo "6" para FAT16 para médias inferiores a 2GB.
  - Tipo "c" para FAT32 (LBA) para médias maiores.
- Formatar a partição primária com o mkfs.vfat(8) com o seguinte.

nome do sistema de ficheiros	cenário de utilização típico
FAT12	partilha de dados em várias plataformas em disquetes (<32MiB)
FAT16	partilha de dados em várias plataformas em dispositivos como pequenos discos rígidos (<2GiB)
FAT32	partilha de dados em várias plataformas em dispositivos como grandes discos rígidos (<8TiB, suportado por mais recente que MS Windows95 OSR2)
exFAT	partilha de dados entre plataformas no dispositivo semelhante a um disco rígido de grandes dimensões (<512TiB, suportado pelo WindowsXP, Mac OS X Snow Leopard 10.6.5 e kernel Linux desde a versão 5.4)
NTFS	partilha de dados em várias plataformas em dispositivos como grandes discos rígidos (suportado nativamente no MS Windows NT e versões posteriores e suportado pelo NTFS-3G via FUSE em Linux)
ISO9660	partilha de dados estáticos em várias plataformas em CD-R e DVD+/-R
UDF	escrita de dados incremental em CD-R e DVD+/-R (novo)
MINIX	armazenamento de dados em ficheiros unix eficiente em espaço em disquetes
ext2	partilha de dados em dispositivos tipo disco rígido com sistemas Linux mais antigos
ext3	partilha de dados em dispositivos tipo disco rígido com sistemas Linux mais antigos
ext4	partilha de dados em dispositivos de tipo disco rígido com sistemas Linux atuais
btrfs	partilha de dados em dispositivos de tipo disco rígido com sistemas Linux atuais com instantâneos só de leitura

Tabela 10.3: Lista de hipóteses de sistemas de ficheiros para dispositivos de armazenamento amovíveis com cenários de utilização típica

- Apenas o nome de dispositivo dele, ex. "`/dev/sda1`" para FAT16
- A opção explícita e o nome de dispositivo dela, ex. "`-F 32 /dev/sda1`" para FAT32

Quando se usa sistemas de ficheiros FAT ou ISO9660 para partilhar dados, as considerações de segurança deverão ser as seguintes.

- Arquivar ficheiros para um ficheiro de arquivo primeiro a utilizar o tar(1), ou cpio(1) para reter o nome longo do ficheiro, a ligação simbólica, as permissões originais de ficheiro Unix e a informação do dono.
- Dividir o ficheiro de arquivo em fatias com menos de 2 GiB com o comando split(1) para o proteger contra limites de tamanho de ficheiro.
- Encriptar o ficheiro de arquivo para segurar o conteúdo dele contra acesso não autorizado.

Nota
Para o sistema de ficheiros FAT pelo seu desenho, o tamanho máximo de ficheiro é `(2^32 - 1) bytes = (4GiB - 1 byte)`. Para algumas aplicações do antigo SO de 32 bits, o tamanho máximo de ficheiro é mais pequeno `(2^31 - 1) bytes = (2GiB - 1 byte)`. O Debian não sofre do segundo problema.

Nota
A própria Microsoft não recomenda o uso de FAT para discos ou partições maiores que 200 MB. A Microsoft destaca as suas deficiências como a ser a utilização ineficiente do espaço do disco na "Visão geral dos sistemas de ficheiros FAT, HPFS e NTFS" dele. Claro que devemos normalmente usar o sistema de ficheiros ext4 para Linux.

Dica
Para mais sistemas de ficheiros e acesso a sistemas de ficheiros, por favor leia "HOWTO dos Sistemas de Ficheiros".

10.1.9 Partilhar dados via a rede

Quando se partilha dados com outro sistema via rede, deve usar serviços comuns. Aqui estão algumas dicas.

serviço de rede	descrição do cenário de utilização típico
SMB/CIFS sistema de ficheiros montado em rede com o Samba	partilha ficheiros via ”Rede Microsoft Windows”, veja smb.conf(5) e O HOWTO Oficial do Samba 3.x.x e Guia de Referência ou o pacote `samba-doc`
NFS sistema de ficheiros montado em rede com o kernel do Linux	partilhar ficheiros via ”Rede Unix/Linux”, veja exports(5) e Linux NFS-HOWTO
serviço HTTP	a partilhar ficheiros entre o servidor/cliente web
serviço HTTPS	partilhar ficheiros entre o servidor/cliente web com Secure Sockets Layer encriptado (SSL) ou Transport Layer Security (TLS)
serviço FTP	a partilhar ficheiros entre o servidor/cliente FTP

Tabela 10.4: Lista de serviços de rede para escolher com o cenário de utilização típico

Apesar de estes sistemas de ficheiros montados sobre rede e métodos de transferência de ficheiros em rede serem bastante convenientes para partilhar dados, estes podem ser inseguros. A ligação de rede deles tem de ser segurada com o seguinte.

- Encriptar com SSL/TLS
- Ligue-o em túnel via SSH
- Ligue-o em túnel via VPN
- Limitar por detrás da firewall segura

Veja também Secção 6.5 e Secção 6.6.

10.2 Salvaguarda (backup) e recuperação

Todos sabemos que os computadores avariam ou que erros humanos causam danos no sistema e nos dados. As operações de salvaguarda e recuperação são a parte essencial sucesso do administrador de sistemas. Todos os modos de falha possíveis irão atingi-lo um dia.

Dica
Mantenha o seu sistema de backup simples e faça backups periódicos. Ter cópias de segurança dos dados é mais importante do que quão bom é tecnicamente o seu método de backup.

10.2.1 Política de cópia de segurança e recuperação

Existem 3 factores chave que determinam a política atual de salvaguarda e recuperação.

1. Saber o que salvaguardar e recuperar.
   - Ficheiros de dados criados directamente por si: dados em ”`~/`”
   - Ficheiros de dados criados por aplicações usadas por si: dados em ”`/var/`” (excepto ”`/var/cache/`”, ”`/var/run/`” e ”`/var/tmp/`”)

- Ficheiros de configuração do sistema: dados em "`/etc/`"
- Programas locais: dados em "`/usr/local/`" ou "`/opt/`"
- Informação da instalação do sistema: um memo em texto simples em passos chave (partição, ...)
- Conjunto de dados de prova: confirmado com antecedência por operações de recuperação experimentais
  - Tarefa cron como um processo de utilizador: ficheiros no diretório "`/var/spool/cron/crontabs`" e reiniciar o cron(8). Veja Secção 9.4.14 para cron(8) e crontab(1).
  - Tarefas de temporizador do Systemd como processos de utilizador: ficheiros no diretório "`~/.config/systemd/`". Veja systemd.timer(5) e systemd.service(5).
  - Trabalhos de arranque automático como processos do utilizador: ficheiros no diretório "`~/.config/autostart`". Consulte Especificação de Início Automático de Aplicações de Ambiente de Trabalho.

2. Saber como salvaguardar e recuperar.
   - Armazenamento de dados seguro: protecção contra reescrita e falha do sistema
   - Salvaguarda frequente: salvaguarda agendada
   - Backup redundante: usar mirror de dados
   - Processo à prova de tolos: backup fácil de comando único
3. Avaliar os riscos e custos envolvidos.
   - Risco de perda de dados
     - Os dados devem estar, pelo menos, em partições de disco diferentes, de preferência em discos e máquinas diferentes, para resistir à corrupção do sistema de ficheiros. É preferível armazenar os dados importantes num sistema de ficheiros só de leitura. [1]
   - Risco de violação de dados
     - Dados de identidade sensíveis, como "`/etc/ssh/ssh_host_*_key`", "`~/.gnupg/*`", "`~/.ssh/*`", "`~/.local`", "`/etc/passwd`", "`/etc/shadow`", "`popularity-contest.conf`", "`/etc/ppp/pap-secrets`", e "`/etc/exim`" devem ser copiados como encriptados. [2] (Ver Secção 9.9.)
     - Nunca codifique a palavra-passe de início de sessão do sistema nem a frase-passe de desencriptação em qualquer script, mesmo em qualquer sistema fiável. (Ver Secção 10.3.6.)
   - Modo de falha e a possibilidade dele
     - O hardware (especialmente o disco rígido) avaria-se
     - O sistema de ficheiros pode estar corrompido e os dados nele contidos podem perder-se
     - O sistema de armazenamento remoto não é fiável em termos de violações de segurança
     - Uma proteção por palavra-passe fraca pode ser facilmente comprometida
     - O sistema de permissão de ficheiros pode estar comprometido
   - Recursos necessários para o backup: humano, hardware, software,…
     - Cópia de segurança automática agendada com a tarefa cron ou a tarefa de temporizador systemd

---

Dica
Pode recuperar dados configuração debconf com "`debconf-set-selections debconf-selections`" e dados de seleção do dpkg com "`dpkg --set-selection <dpkg-selections.list`".

Nota
Não faça salvaguarda aos conteúdos dos pseudo-sistemas de ficheiros encontrados em `/proc`, `/sys`, `/tmp` e `/run` (veja Secção 1.2.12 e Secção 1.2.13). A menos que saiba exatamente o que está a fazer, eles são enormes quantidades de dados desnecessários.

---

1 Uma média de escrita única como o CD/DVD-R pode evitar acidentes de sobreescrita. (Veja Secção 9.8 para saber como escrever para o suporte de armazenamento a partir da linha de comandos da shell. O ambiente GUI do ambiente de trabalho GNOME dá-lhe acesso fácil através do menu: "Locais → Criador de CD/DVD".)

2 Alguns destes dados não podem ser gerados de novo introduzindo a mesma cadeia de entrada no sistema.

Nota
Pode desejar parar alguns daemons de aplicação como o MTA (veja Secção 6.2.4) enquanto faz cópias de segurança (backups) dos dados.

10.2.2 Suites de utilitários de backup

Aqui está uma lista selcionada de suites de utilitários de backup notáveis disponíveis no sistema Debian.

pacote	popcon	tamanho	descrição
bacula-common	V:6.0, I:6.7	2501	Bacula: salvaguarda, recuperação e verificação em rede - ficheiros de suporte comum
bacula-client	V:0.2, I:1.8	199	Bacula: salvaguarda, recuperação e verificação em rede - meta-pacote cliente
bacula-console	V:0.6, I:2.1	112	Bacula: salvaguarda, recuperação e verificação em rede - consola de texto
bacula-server	I:0.59	199	Bacula: salvaguarda, recuperação e verificação em rede - meta-pacote servidor
amanda-common	V:0.6, I:2.1	9951	Amanda: Advanced Maryland Automatic Network Disk Archiver (Bibliotecas)
amanda-client	V:0.5, I:2.1	1109	Amanda: Advanced Maryland Automatic Network Disk Archiver (Cliente)
amanda-server	V:0.11, I:0.22	1095	Amanda: Advanced Maryland Automatic Network Disk Archiver (Servidor)
backuppc	V:1.4, I:1.6	3088	BackupPC é um sistema de grau empresarial de alta performance para fazer salvaguardas a PCs (baseado em disco)
duplicity	V:5, I:44	2662	salvaguarda incremental (remoto)
deja-dup	V:28, I:41	5274	Interface gráfica para duplicidade
borgbackup	V:13, I:28	3532	Cópia de segurança com desduplicação (remota)
borgmatic	V:3.2, I:4.4	946	auxiliar borgbackup
rdiff-backup	V:2.1, I:6.1	1169	salvaguarda incremental (remoto)
restic	V:8, I:15	41617	salvaguarda incremental (remoto)
backupninja	V:2.0, I:2.4	360	sistema de **meta-backup** leve e extensível
slbackup	V:0.10, I:0.13	147	salvaguarda incremental (remoto)
backup-manager	V:0.41, I:0.70	573	ferramenta de salvaguarda de linha de comandos
backup2l	V:0.35, I:0.46	110	ferramenta de baixa manutenção para salvaguarda/restauro para medias montáveis (baseado em disco)

Tabela 10.5: Lista de suites utilitárias de salvaguarda

As ferramentas de salvaguarda têm os seus objetivos especializados.

- Mondo Rescue é um sistema de backup para facilitar o restauro de um sistema completo rapidamente a partir de CD/DVD, etc de backup, sem se passar por todo o processo normal de instalação do sistema.
- Bacula, Amanda e BackupPC são suites utilitárias de salvaguarda cheias de funcionalidades que se destinam a salvaguardas regulares em rede.
- Duplicity, e Borg são utilitários de backup mais simples para estações de trabalho típicas.

10.2.3 Sugestões de cópia de segurança

Para uma estação de trabalho pessoal, os utilitários de backup completos concebidos para o ambiente de servidor podem não ser úteis. Ao mesmo tempo, os utilitários de backup existentes para estações de trabalho podem ter algumas deficiências.

Seguem-se algumas sugestões para facilitar a cópia de segurança com o mínimo de esforço do utilizador. Estas técnicas podem ser utilizadas com qualquer utilitário de cópia de segurança.

Para fins de demonstração, vamos assumir que o utilizador principal e o nome do grupo são `pinguim` e criar um exemplo de script de cópia de segurança e instantâneo "`/usr/local/bin/bkss.sh`" como:

```
#!/bin/sh -e
SRC="$1" # source data path
DSTFS="$2" # backup destination filesystem path
DSTSV="$3" # backup destination subvolume name
DSTSS="${DSTFS}/${DSTSV}-snapshot" # snapshot destination path
if [ "$(stat -f -c %T "$DSTFS")" != "btrfs" ]; then
  echo "E: $DESTFS needs to be formatted to btrfs" >&2 ; exit 1
fi
MSGID=$(notify-send -p "bkup.sh $DSTSV" "in progress ...")
if [ ! -d "$DSTFS/$DSTSV" ]; then
  btrfs subvolume create "$DSTFS/$DSTSV"
  mkdir -p "$DSTSS"
fi
rsync -aHxS --delete --mkpath "${SRC}/" "${DSTFS}/${DSTSV}"
btrfs subvolume snapshot -r "${DSTFS}/${DSTSV}" ${DSTSS}/$(date -u --iso=min)
notify-send -r "$MSGID" "bkup.sh $DSTSV" "finished!"
```

Aqui, apenas a ferramenta básica rsync(1) é utilizada para facilitar a cópia de segurança do sistema e o espaço de armazenamento é utilizado de forma eficiente pelo Btrfs.

Dica
Para sua informação: Este autor utiliza o seu próprio script de shell semelhante "bss: Btrfs Subvolume Snapshot Utility" para a sua estação de trabalho.

10.2.3.1 Cópia de segurança GUI

Aqui está um exemplo para configurar a cópia de segurança de um único clique da GUI.

- Preparar um dispositivo de armazenamento USB para ser utilizado como cópia de segurança.
  - Formate um dispositivo de armazenamento USB com uma partição em btrfs com o seu nome de etiqueta como "BKUP". Isto pode ser encriptado (ver Secção 9.9.1).
  - Ligue-o ao seu sistema. O sistema da área de trabalho deve montá-lo automaticamente como "`/media/penguin/BKUP`"
  - Execute "`sudo chown penguin:penguin /media/penguin/BKUP`" para que o utilizador possa escrever nele.
- Crie "`~/.local/share/applications/BKUP.desktop`" seguindo as técnicas escritas Secção 9.4.10 como:

```
[Desktop Entry]
Name=bkss
Comment=Backup and snapshot of ~/Documents
Exec=/usr/local/bin/bkss.sh /home/penguin/Documents /media/penguin/BKUP Documents
Type=Application
```

Para cada clique na GUI, é feita uma cópia de segurança dos seus dados de "`~/Documents`" para um dispositivo de armazenamento USB e é criada uma imagem só de leitura.

10.2.3.2 Cópia de segurança acionada por evento de montagem

Eis um exemplo de configuração para a cópia de segurança automática acionada pelo evento de montagem.

- Prepare um dispositivo de armazenamento USB para ser utilizado para a cópia de segurança, como em Secção 10.2.3.1.
- Crie um ficheiro de unidade de serviço systemd "`~/.config/systemd/user/back-BKUP.service`" como:

```
[Unit]
Description=USB Disk backup
Requires=media-%u-BKUP.mount
After=media-%u-BKUP.mount

[Service]
ExecStart=/usr/local/bin/bkss.sh %h/Documents /media/%u/BKUP Documents
StandardOutput=append:%h/.cache/systemd-snap.log
StandardError=append:%h/.cache/systemd-snap.log

[Install]
WantedBy=media-%u-BKUP.mount
```

- Ativar esta configuração da unidade systemd com o seguinte:

```
$ systemctl --user enable bkup-BKUP.service
```

Para cada evento de montagem, é feita uma cópia de segurança dos seus dados de "`~/Documents`" para um dispositivo de armazenamento USB e é criada uma imagem só de leitura.

Aqui, os nomes das unidades montadas systemd, que o systemd tem atualmente em memória podem ser pedidos ao gestor de serviços do utilizador que os chama com "`systemctl --user list-units --type=mount`".

10.2.3.3 Cópia de segurança acionada por um evento de temporizador

Eis um exemplo de configuração da cópia de segurança automática acionada pelo evento do temporizador.

- Prepare um dispositivo de armazenamento USB para ser utilizado para a cópia de segurança, como em Secção 10.2.3.1.
- Crie um ficheiro systemd de unidade temporizadora "`~/.config/systemd/user/snap-Documents.timer`" como:

```
[Unit]
Description=Run btrfs subvolume snapshot on timer
Documentation=man:btrfs(1)

[Timer]
OnStartupSec=30
OnUnitInactiveSec=900

[Install]
WantedBy=timers.target
```

- Crie um ficheiro systemd de unidade de serviço "`~/.config/systemd/utilizador/snap-Documents.service`" como:

```
[Unit]
Description=Run btrfs subvolume snapshot
Documentation=man:btrfs(1)

[Service]
Type=oneshot
Nice=15
ExecStart=/usr/local/bin/bkss.sh %h/Documents /media/%u/BKUP Documents
IOSchedulingClass=idle
CPUSchedulingPolicy=idle
StandardOutput=append:%h/.cache/systemd-snap.log
StandardError=append:%h/.cache/systemd-snap.log
```

- Ativar esta configuração da unidade systemd com o seguinte:

```
$ systemctl --user enable snap-Documents.timer
```

Para cada evento do temporizador, é feita uma cópia de segurança dos seus dados de "`~/Documents`" para um dispositivo de armazenamento USB e é criada uma imagem só de leitura.

Aqui, os nomes das unidades temporizadoras de utilizador do systemd, que o systemd tem atualmente em memória podem ser perguntados ao gestor de serviços do utilizador que os chama com "`systemctl --user list-units --type=timer`".

Para o sistema de área de trabalho moderna, esta abordagem do systemd pode oferecer um controlo mais fino do que os tradicionais sistemas Unix que utilizam at(1), cron(8), ou anacron(8).

10.3 Infraestrutura da segurança de dados

A infraestrutura de segurança dos dados é disponibilizada pela combinação de uma ferramenta de encriptação de dados, ferramenta de resumo de mensagens e ferramenta de assinaturas.

Veja Secção 9.9 em dm-crypt e fscrypt os quais implementam infraestruturas de encriptação de dados automática via módulos do kernel Linux.

10.3.1 Gestão de chaves para GnuPG

Aqui estão comandos do GNU Privacy Guard para gestão de chaves básica.

Aqui está o significado do código de confiança.

O seguinte gera a minha chave "`9563FC29932C409F1A77F9C1AB8A1522D8234C6A`".

```
$ gpg --gen-key
gpg (GnuPG) 2.4.7; Copyright (C) 2024 g10 Code GmbH
...
GnuPG needs to construct a user ID to identify your key.

Real name: Osamu Aoki
Email address: osamu@debian.org
You selected this USER-ID:
    "Osamu Aoki <osamu@debian.org>"

Change (N)ame, (E)mail, or (O)kay/(Q)uit? o
...
public and secret key created and signed.
```

pacote	popcon	tamanho	comando	descrição
gnupg	V:418, I:860	465	gpg(1)	GNU Privacy Guard - Ferramenta de encriptação e assinatura OpenPGP
gpgv	V:203, I:923	577	gpgv(1)	GNU Privacy Guard - Ferramenta de verificação de assinaturas OpenPGP
sq	V:1, I:33	24447	sq(1)	Sequoia-PGP - Ferramenta de encriptação e assinatura OpenPGP
sqv	V:479, I:554	1919	sqv(1)	Sequoia-PGP - Ferramenta de verificação de assinaturas OpenPGP
paperkey	V:1, I:12	58	paperkey(1)	extrai apenas a informação secreta de chaves secretas OpenPGP
cryptsetup	V:34, I:77	469	cryptsetup(8), …	utilitários para encriptação de dispositivos de bloco dm-crypt com suporte de LUKS
coreutils	V:912, I:1000	17994	md5sum(1)	computa e verifica o resumo da mensagem MD5
coreutils	V:912, I:1000	17994	sha1sum(1)	computa e verifica o resumo da mensagem SHA1
openssl	V:846, I:996	2564	openssl(1ssl)	computa o resumo da mensagem com ”openssl dgst” (OpenSSL)
libsecret-tools	V:2, I:13	44	secret-tool(1)	guardar e recuperar palavras-passe (CLI)
seahorse	V:76, I:260	7999	seahorse(1)	ferramenta de gestão de chaves (GNOME)

Tabela 10.6: Lista de ferramentas de infraestrutura da segurança de dados

comando	descrição
gpg --gen-key	gerar uma chave nova
gpg --gen-revoke my_user_ID	gera chave de revogação para meu_ID_utilizador
gpg --edit-key user_ID	edita chave interativamente, ”help” para ajuda
gpg -o file --export	exporta todas as chaves públicas para ficheiro
gpg -o file --export-secret-keys	exporta todas as chaves privadas para ficheiro
gpg --import file	importa todas as chaves de ficheiro
gpg --send-keys user_ID	envia chave de ID_utilizador para servidor de chaves
gpg --recv-keys user_ID	recupera chave de ID_utilizador do servidor de chaves
gpg --list-keys user_ID	lista chaves de ID_utilizador
gpg --list-sigs user_ID	lista assinaturas de ID_utilizador
gpg --check-sigs user_ID	verifica assinaturas de ID_utilizador
gpg --fingerprint user_ID	verifica a impressão digital de ID_utilizador
gpg --refresh-keys	atualiza o chaveiro local

Tabela 10.7: Lista de comandos do GNU Privacy Guard para gestão de chaves

código	descrição de confiança
-	nenhuma confiança de dono atribuída / ainda não calculado
e	falha no cálculo da confiança
q	não existe informação suficiente para o cálculo
n	nunca confiar nesta chave
m	marginalmente confiável
f	totalmente confiável
u	de confiança absoluta

Tabela 10.8: Lista do significado do código de confiança

```
pub   ed25519 2026-02-14 [SC] [expires: 2029-02-13]
      9563FC29932C409F1A77F9C1AB8A1522D8234C6A
uid                      Osamu Aoki <osamu@debian.org>
sub   cv25519 2026-02-14 [E] [expires: 2029-02-13]
```

O seguinte envia a minha chave "9563FC29932C409F1A77F9C1AB8A1522D8234C6A" para o popular servidor de chaves "hkp://keys.gnupg.net".

```
$ gpg --keyserver hkp://keys.gnupg.net --send-keys 9563FC29932C409F1A77F9C1AB8A1522D8234C6A
```

Um bom servidor de chaves predefinido configurado em "~/.gnupg/gpg.conf" (ou na antiga localização "~/.gnupg/opt
contém o seguinte.

```
keyserver hkp://keys.gnupg.net
```

O seguinte obtém chaves desconhecidas do servidor de chaves.

```
$ gpg --list-sigs --with-colons | grep '^sig.*\[User ID not found\]' |\
          cut -d ':' -f 5| sort | uniq | xargs gpg --recv-keys
```

Existiu um bug no OpenPGP Public Key Server (versão anterior a 0.9.6) que corrompeu as chaves com mais de 2 sub-chaves. O novo pacote gnupg (>1.2.1-2) consegue lidar com estas chaves corrompidas. Veja gpg(1) sob a opção "--repair-pks-subkey-bug".

O uso de SHA-1 para hash foi descontinuado. Se a sua chave OpenPGP antiga baseada em RSA usa SHA-1 para hash, correja-a usando FixKeyWithSha1.

Dica
Os comandos sq(1) e sqv(1) são um conjunto alternativo de comandos openPGP. Veja documentação do utilizador do sq e Uma Introdução Prática para Usar sq, CLI do Sequoia PGP.

10.3.2 Usa GnuPG em ficheiros

Aqui estão exemplos para usar comandos do GNU Privacy Guard em ficheiros.

10.3.3 Usar GnuPG com o Mutt

Adicione o seguinte a "~/.muttrc" para impedir o GnuPG lento de arrancar automaticamente, enquanto permite que seja usado ao escrever "S" no menu de índice.

```
macro index S ":toggle pgp_verify_sig\n"
set pgp_verify_sig=no
```

10.3.4 Usar GnuPG com o Vim

O plugin gnupg permite correr GnuPG de forma transparente para ficheiros com extensão ".gpg", ".asc" e ".pgp".[3]

```
$ sudo aptitude install vim-scripts
$ echo "packadd! gnupg" >> ~/.vim/vimrc
```

3Se utilizar "~/.vimrc" em vez de "~/.vim/vimrc", substitua-o em conformidade.

comando	descrição
`gpg -a -s file`	assina ficheiro em ficheiro.asc blindado de ASCII
`gpg --armor --sign file`	,,
`gpg --clearsign file`	mensagem com assinatura clara
`gpg --clearsign file|mail foo@example.org`	envia por mail uma mensagem com assinatura clara para `foo@example.org`
`gpg --clearsign --not-dash-escaped patchfile`	ficheiro patch com assinatura clara
`gpg --verify file`	verifica ficheiro com assinatura clara
`gpg -o file.sig -b file`	cria assinatura separada
`gpg -o file.sig --detach-sign file`	,,
`gpg --verify file.sig file`	verifica ficheiro com file.sig
`gpg -o crypt_file.gpg -r name -e file`	encriptação de chave pública destinada a nome a partir de ficheiro para crypt_file.gpg binário
`gpg -o crypt_file.gpg --recipient name --encrypt file`	,,
`gpg -o crypt_file.asc -a -r name -e file`	encriptação de chave pública destinada a nome a partir de ficheiro para crypt_file.asc blindado de ASCII
`gpg -o crypt_file.gpg -c file`	encriptação simétrica a partir de ficheiro para crypt_file.gpg
`gpg -o crypt_file.gpg --symmetric file`	,,
`gpg -o crypt_file.asc -a -c file`	encriptação simétrica destinada a nome a partir de ficheiro para crypt_file.asc blindado de ASCII
`gpg -o file -d crypt_file.gpg -r name`	desencriptação
`gpg -o file --decrypt crypt_file.gpg`	,,

Tabela 10.9: Lista de comandos do GNU Privacy Guard em ficheiros

10.3.5 O valor de controlo MD5

O md5sum(1) disponibiliza um utilitário para fazer um ficheiro de sumário a usar o método em rfc1321 e verificar cada ficheiro com ele.

```
$ md5sum foo bar >baz.md5
$ cat baz.md5
d3b07384d113edec49eaa6238ad5ff00  foo
c157a79031e1c40f85931829bc5fc552  bar
$ md5sum -c baz.md5
foo: OK
bar: OK
```

Nota
O cálculo do sumário MD5 é menos intensivo para a CPU que o da assinatura criptográfica do GNU Privacy Guard (GnuPG). Normalmente, apenas o ficheiro de digestão do nível de topo é assinado criptograficamente para assegurar a integridade dos dados.

10.3.6 Gestor de palavras-passe

No sistema GNOME, a ferramenta GUI seahorse(1) gere as palavras-passe e guarda-as de forma segura no chaveiro `~/.local/share/keyrings/*`.

secret-tool(1) pode armazenar a palavra-passe no chaveiro a partir da linha de comandos.

Vamos armazenar a frase-chave utilizada para a imagem de disco encriptada LUKS/dm-crypt

```
$ secret-tool store --label='LUKS passphrase for disk.img' LUKS my_disk.img
Password: ********
```

Esta palavra-passe armazenada pode ser recuperada e fornecida a outros programas, por exemplo, cryptsetup(8).

```
$ secret-tool lookup LUKS my_disk.img | \
  cryptsetup open disk.img disk_img --type luks --keyring -
$ sudo mount /dev/mapper/disk_img /mnt
```

Dica
Sempre que precisar de fornecer uma palavra-passe num script, utilize a `ferramenta-secreta` e evite codificar diretamente a palavra-passe no script.

10.4 Ferramentas de fusão de código fonte

Existem muitas ferramentas de fusão para código fonte. Os seguinte comandos chamaram a minha atenção.

10.4.1 Extrair as diferenças para ficheiros fonte

Os seguintes procedimentos extraem as diferenças entre dois ficheiros de fonte e cria os ficheiros diff unificados "`file.patch0`" ou "`file.patch1`" a depender da localização do ficheiro.

```
$ diff -u file.old file.new > file.patch0
$ diff -u old/file new/file > file.patch1
```

pacote	popcon	tamanho	comando	descrição
patch	V:102, I:727	242	patch(1)	aplica ficheiro diff a um original
vim	V:83, I:338	4164	vimdiff(1)	compara dois ficheiros lado a lado no vim
imediff	V:0.02, I:0.25	329	imediff(1)	ferramenta de fusão de 2 ou 3 vias interactiva de écran completo
meld	V:5, I:23	3996	meld(1)	compara e funde ficheiros (GTK)
wiggle	V:0.02, I:0.12	175	wiggle(1)	aplica patches rejeitadas
diffutils	V:896, I:998	1768	diff(1)	compara ficheiros linha a linha
diffutils	V:896, I:998	1768	diff3(1)	compara e junta três ficheiros linha a linha
quilt	V:2, I:16	880	quilt(1)	gere séries de patches
wdiff	V:5, I:38	580	wdiff(1)	mostra diferenças de palavras entre ficheiros de texto
diffstat	V:9, I:93	79	diffstat(1)	produz um histograma de alterações feitas pelo diff
patchutils	V:11, I:92	269	combinediff(1)	cria uma patch cumulativa de duas patches incrementais
patchutils	V:11, I:92	269	dehtmldiff(1)	extrai um diff de uma página HTML
patchutils	V:11, I:92	269	filterdiff(1)	extrai ou executa diffs de um ficheiro diff
patchutils	V:11, I:92	269	fixcvsdiff(1)	corrige ficheiros diff criados pelo CVS que o patch(1) interpreta mal
patchutils	V:11, I:92	269	flipdiff(1)	troca a ordem de duas patches
patchutils	V:11, I:92	269	grepdiff(1)	mostra que ficheiros são modificados por uma patch que corresponde a um regex
patchutils	V:11, I:92	269	interdiff(1)	mostra as diferenças entre dois ficheiros diff unificados
patchutils	V:11, I:92	269	lsdiff(1)	mostra quais ficheiros são modificados por uma patch
patchutils	V:11, I:92	269	recountdiff(1)	recalcula contagens e offsets em diffs de contexto unificado
patchutils	V:11, I:92	269	rediff(1)	corrige os offsets e as contagens de um diff editado manualmente
patchutils	V:11, I:92	269	splitdiff(1)	separa patches incrementais
patchutils	V:11, I:92	269	unwrapdiff(1)	desembaralha patches cujas linhas foram alteradas para arrumação de palavras
dirdiff	V:0.1, I:1.3	167	dirdiff(1)	mostra diferenças e funde alterações entre árvores de diretórios
docdiff	V:0.01, I:0.16	554	docdiff(1)	compara dois ficheiros palavra a palavra / caractere a caractere
makepatch	V:0.01, I:0.14	99	makepatch(1)	gera ficheiros de patch extensos
makepatch	V:0.01, I:0.14	99	applypatch(1)	aplica ficheiros de patch extensos

Tabela 10.10: Lista de ferramentas de fusão de código fonte

10.4.2 Fundir atualizações para ficheiros de fonte

O ficheiro diff (alternativamente chamado ficheiro patch) é usado para enviar uma atualização de um programa. A parte receptora aplica esta atualização a outro ficheiro com o seguinte.

```
$ patch -p0 file < file.patch0
$ patch -p1 file < file.patch1
```

10.4.3 Integração interativa

Se tiver duas versões de um código fonte, pode executar uma fusão-de-2-vias efetiva a usar o imediff(1) com o seguinte.

```
$ imediff -o file.merged file.old file.new
```

Se tiver três versões de um código fonte, pode executar uma fusão de 3-vias efetiva ao usar o imediff(1) com o seguinte.

```
$ imediff -o file.merged file.yours file.base file.theirs
```

10.5 Git

Atualmente, o Git é a ferramenta de eleição para o sistema de controlo de versões (VCS), uma vez que o Git pode fazer tudo para a gestão local e remota do código-fonte.

O Debian disponibiliza serviços Git livres via Serviço Debian Salsa.. A sua documentação encontra-se em https://wiki.debian.Salsa .

Aqui estão alguns pacotes relacionados com o Git.

10.5.1 Configuração do cliente Git

Pode desejar definir várias configurações globais em "`~/.gitconfig`" como o seu nome e endereço de mail usado pelo Git com o seguinte.

```
$ git config --global user.name "Name Surname"
$ git config --global user.email yourname@example.com
```

Também pode personalizar o comportamento predefinido do Git da seguinte forma.

```
$ git config --global init.defaultBranch main
$ git config --global pull.rebase true
$ git config --global push.default current
```

Se está muito acostumado aos comandos do CVS ou Subversion, pode desejar definir nomes alternativos a vários comandos com o seguinte.

```
$ git config --global alias.ci "commit -a"
$ git config --global alias.co checkout
```

Pode verificar a sua configuração global com o seguinte.

```
$ git config --global --list
```

pacote	popcon	tamanho	comando	descrição
git	V:422, I:625	52512	git(7)	Git, o sistema de controlo de revisão distribuído, rápido e escalável
gitk	V:3, I:26	2045	gitk(1)	explorador GUI de repositórios Git com histórico
qgit	V:0.2, I:1.9	1431	qgit(1)	explorador GUI de repositórios Git com histórico
git-cola	V:0.8, I:4.5	5129	git-cola(1)	explorador GUI de repositórios Git com histórico
tig	V:2, I:11	1245	tig(1)	explorador de repositórios Git de consola com histórico
lazygit	V:1.0, I:4.1	24167	lazygit(1)	explorador de repositórios Git de consola com histórico
git-gui	V:1, I:16	2549	git-gui(1)	GUI para Git (Nenhum histórico)
git-email	V:0.3, I:9.3	1229	git-send-email(1)	envia uma colecção de patches como email a partir do Git
git-buildpackage	V:1.2, I:6.8	2027	git-buildpackage(1)	automatiza o empacotamento Debian com o Git
dgit	V:0.12, I:0.97	757	dgit(1)	interoperabilidade do git com o repositório Debian
imediff	V:0.02, I:0.25	329	git-ime(1)	ferramenta auxiliar interativa de divisão de commits do git
stgit	V:0.08, I:0.50	6253	stg(1)	quilt no topo do git (Python)
git-doc	I:10	15192	N/D	documentação oficial para o Git
gitmagic	I:0.48	721	N/D	”Magia do Git”, guia fácil de compreender para o Git

Tabela 10.11: Lista de pacotes e comandos relacionados com o git

10.5.2 Comandos básicos do Git

A operação Git envolve vários dados.

- A árvore de trabalho que contém os ficheiros que o utilizador enfrenta e nos quais faz alterações.
  - As alterações a serem registadas devem ser explicitamente selecionadas e colocadas no índice. Estes são os comandos `git add` e `git rm`.
- O índice que contém os ficheiros preparados.
  - Os ficheiros preparados serão confirmados para o repositório local após o pedido subsequente. Este é o comando `git commit`.
- O repositório local que contém os ficheiros confirmados.
  - O Git regista o histórico ligado dos dados confirmados e organiza-os como ramos no repositório.
  - O repositório local pode enviar dados para o repositório remoto através do comando `git push`.
  - O repositório local pode receber dados do repositório remoto através dos comandos `git fetch` e `git pull`.
    - O comando `git pull` executa o comando `git merge` ou `git rebase` após o comando `git fetch`.
    - Aqui, `git merge` combina dois ramos separados do histórico no final para um ponto. (Este é o padrão do `git pull` sem personalização e pode ser bom para as pessoas que publicam o ramo para muitas pessoas.)
    - Aqui, `git rebase` cria um único ramo do histórico sequencial do ramo remoto seguido pelo ramo local. (Este é o caso de personalização `pull.rebase true` e pode ser bom para o resto de nós.)
- O repositório remoto que contém os ficheiros confirmados.
  - A comunicação com o repositório remoto utiliza protocolos de comunicação seguros, como SSH ou HTTPS.

A árvore de trabalho são os arquivos fora do diretório `.git/`. Os ficheiros dentro do diretório `.git/` contêm o índice, os dados do repositório local e alguns ficheiros de texto de configuração do git.

Aqui está uma visão geral dos principais comandos do Git.

Comando Git	função
`git init`	cria o repositório (local)
`git clone URL`	clonar o repositório remoto para um repositório local com a árvore de trabalho
`git pull origin main`	atualizar o ramo `main` local pela `origin` do repositório remoto
`git add .`	adicionar ficheiro(s) na árvore de trabalho ao índice para ficheiros pré-existentes apenas no índice
`git add -A .`	adicionar ficheiro(s) na árvore de trabalho ao índice para todos os ficheiros, incluindo remoções
`git rm filename`	remover ficheiro(s) da árvore de trabalho e do índice
`git commit`	confirmar as alterações preparadas no índice para o repositório local
`git commit -a`	adicionar todas as alterações da árvore de trabalho ao índice e submetê-las ao repositório local (adicionar + submeter)
`git push -u origin branch_name`	atualizar a `origem` do repositório remoto pelo ramo local `nome_ramo` (invocação inicial)
`git push origin branch_name`	atualizar a `origem` do repositório remoto pelo ramo local `nome_ramo` (invocação subsequente)
`git diff treeish1 treeish2`	mostrar a diferença entre o envio *treeish1* e o envio *treeish2*
`gitk`	Exibição GUI da árvore de histórico de ramificações do repositório VCS

Tabela 10.12: Principais comandos do Git

10.5.3 Dicas do Git

Eis algumas dicas do Git.



Atenção
Não use a string de etiqueta (tag) com espaços nela, mesmo que algumas ferramentas como o gitk(1) o permitam. Pode estrangular outros comandos do `git`.



Cuidado
Se um ramo local que foi enviado para o repositório remoto for rebaseado ou squashed, o envio deste ramo tem riscos e requer a opção `--force`. Isso geralmente não é aceitável para o ramo `main`, mas pode ser aceitável para um ramo tópico antes de mesclar com o ramo `main`.



Cuidado
Invocar um sub-comando `git` directamente como ”`git-xyz`” a partir da linha de comandos foi descontinuado desde o inicio de 2006.

Dica
Se existir um ficheiro executável `git-foo` no caminho especificado por `$PATH`, inserir ”`git foo`” sem hífen na linha de comandos invoca este `git-foo`. Isto é uma característica do comando `git`.

10.5.4 Referências do Git

Veja o seguinte.

Linha de comando do Git	função
`gitk --all`	ver o histórico completo do Git e operar sobre ele, como redefinir o HEAD para outro commit, escolher patches, criar tags e branches ...
`git stash`	obter a árvore de trabalho limpa sem perder dados
`git remote -v`	verificar as definições do controlo remoto
`git branch -vv`	verificar definições para o seu ramo
`git status`	mostrar o estado da árvore de trabalho
`git config -l`	listar definições git
`git reset --hard HEAD; git clean -x -d -f`	reverter todas as alterações da árvore de trabalho e limpá-las completamente
`git rm --cached filename`	reverte o índice de fases alterado pelo `git add filename`
`git reflog`	obter registo de referência (útil para recuperar commits do ramo removido)
`git branch new_branch_name HEAD@{6}`	criar um novo ramo a partir da informação do reflog
`git remote add new_remote URL`	adicionar um repositório remoto `new_remote` apontado pelo URL
`git remote rename origin upstream`	renomear o nome do repositório remoto de `origin` para `upstream`
`git branch -u upstream/branch_name`	defina o rastreio remoto para o repositório remoto `upstream` e o seu nome de ramo `branch_name`.
`git remote set-url origin https://foo/bar.git`	alterar URL de `origin`
`git remote set-url --push upstream DISABLED`	desativar o envio para `upstream` (Editar `.git/config` para voltar a ativar)
`git remote update upstream`	obter atualizações de todos os ramos remotos no repositório "`upstream`" a montante
`git fetch upstream foo:upstream-foo`	criar um ramo `upstream-foo` local (possivelmente órfão) como uma cópia do ramo `foo` no repositório `upstream`
`git checkout -b topic_branch ; git push -u topic_branch origin`	criar um novo `tópico_de-ramo` e enviá-lo para a `origem`
`git branch -m oldname newname`	renomear o nome do ramo local
`git push -d origin branch_to_be_removed`	remover o ramo remoto (novo método)
`git push origin :branch_to_be_removed`	remover o ramo remoto (método antigo)
`git checkout --orphan unconnected`	criar um novo ramo `unconnected`
`git rebase -i origin/main`	reorder/drop/squish commits de `origin/main` para limpar o histórico do ramo
`git reset HEAD^; git commit --amend`	squash os últimos 2 commits num só
`git checkout topic_branch ; git merge --squash topic_branch`	squash todo o `topic_branch` num commit
`git fetch --unshallow --update-head-ok origin '+refs/heads/*:refs/heads/*'`	converter um clone superficial num clone completo de todos os ramos
`git ime`	divide o último commit numa série de commits mais pequenos, ficheiro a ficheiro, etc. (é necessário o pacote `imediff`)
`git repack -a -d; git prune`	reempacotar o repositório local num único pacote (isto pode limitar a possibilidade de recuperação de dados perdidos a partir de um ramo apagado, etc.)

Tabela 10.13: Dicas do Git

- manual: git(1) (`/usr/share/doc/git-doc/git.html`)
- Manual do Utilizador do Git (`/usr/share/doc/git-doc/user-manual.html`)
- Um tutorial de introdução ao git (`/usr/share/doc/git-doc/gittutorial.html`)
- Um tutorial de introdução ao git: parte dois (`/usr/share/doc/git-doc/gittutorial-2.html`)
- GIT do Dia-a-Dia com cerca de 20 comandos (`/usr/share/doc/git-doc/giteveryday.html`)
- Magia do Git (`/usr/share/doc/gitmagic/html/index.html`)

10.5.5 Outros sistemas de controlo de versões

O sistema de controlo de versões (VCS) é por vezes conhecido como sistema de controlo de revisões (RCS) ou gestão da configuração do software (SCM).

Aqui está um sumário dos outros VCS não-Git notáveis no sistema Debian.

pacote	popcon	tamanho	ferramenta	Tipo VCS	comentário
`mercurial`	V:3, I:25	1375	Mercurial	distribuído	DVCS em Python e algum C
`darcs`	V:0.2, I:3.1	38856	Darcs	distribuído	DVCS com álgebra inteligente de patches (lento)
`tla`	V:0.03, I:0.67	1022	GNU arch	distribuído	DVCS principalmente por Tom Lord (histórico)
`bzr`	V:0.0, I:3.9	28	GNU Bazaar	distribuído	DVCS influenciado pelo `tla` escrito em Python (histórico)
`subversion`	V:9, I:54	4856	Subversion	remoto	”CVS feito corretamente”, o mais recente VCS remoto padrão (histórico)
`cvs`	V:3, I:23	4835	CVS	remoto	VCS remoto padrão anterior (histórico)
`tkcvs`	V:0.11, I:0.83	34	CVS, …	remoto	ecrã GUI de árvores de repositório VCS (CVS, Subversion, RCS)
`rcs`	V:1.6, I:9.0	578	RCS	local	”SCCS Unix bem feito” (histórico)
`cssc`	V:0.01, I:0.28	2044	CSSC	local	clone do SCCS Unix (histórico)

Tabela 10.14: Lista de outras ferramentas de sistemas de controlo de versões

Capítulo 11

Conversão de dados

São descritas ferramentas e dicas para converter formatos de dados no sistema Debian.

As ferramentas baseadas em standards são muitos boas mas o suporte para formatos proprietários de dados é limitado.

11.1 Ferramentas de conversão de dados em texto

Os seguintes pacotes para a conversão de dados de texto saltaram-me à vista.

pacote	popcon	tamanho	palavra chave	descrição
libc6	V:936, I:1000	5483	conjunto e caracteres (charset)	converter codificação de texto entre locales por iconv(1) (fundamental)
recode	V:2, I:12	528	conjunto de caracte-res+eol	conversor de codificação de texto entre locales (versátil, com mais nomes alternativos (alias) e funcionalidades)
konwert	V:2, I:42	137	conjunto e caracteres (charset)	conversor de codificação de texto entre locales (imaginativo)
nkf	V:0.4, I:8.6	360	conjunto e caracteres (charset)	tradutor de conjunto de caracteres para Japonês
tcs	V:0.01, I:0.14	518	conjunto e caracteres (charset)	tradutor de conjunto de caracteres
unaccent	V:0.03, I:0.27	35	conjunto e caracteres (charset)	substitui letras acentuadas pelo seu equivalente não acentuado
tofrodos	V:1, I:12	50	eol	conversor de formato de texto entre DOS e Unix: `de dos`(1) e `para dos`(1)
macutils	V:0.04, I:0.39	319	eol	conversor de formato de texto entre Macintosh e Unix: `de mac`(1) e `para mac`(1)

Tabela 11.1: Lista de ferramentas de conversão de dados em texto

11.1.1 Converter um ficheiro de texto com o iconv

Dica
iconv(1) é disponibilizado como parte do pacote `libc6` e está sempre disponível em praticamente todos os sistemas tipo Unix para converter a codificação de caracteres.

Pode converter a codificação de um ficheiro de texto com o iconv(1) com o seguinte.

```
$ iconv -f encoding1 -t encoding2 input.txt >output.txt
```

Os valores de codificação são sensíveis a maiúsculas/minúsculas e ignoram "`-`" e "_" para correspondência. As codificações suportadas podem ser verificadas pelo comando "`iconv -l`".

valor de codificação	utilização
ASCII	American Standard Code for Information Interchange, código de 7 bits sem caracteres acentuados
UTF-8	standard multilingue atual para todos os sistemas operativos modernos
ISO-8859-1	antigo standard para linguagens da Europa ocidental, ASCII + caracteres acentuados
ISO-8859-2	antigo standard para linguagens da Europa oriental, ASCII + caracteres acentuados
ISO-8859-15	antigo standard para linguagens da Europa ocidental, o ISO-8859-1 com o símbolo do euro
CP850	página de código 850, caracteres DOS da Microsoft com gráficos para linguagens da Europa ocidental, variante ISO-8859-1
CP932	página de código 932, variante Shift-JIS do estilo Microsoft Windows para Japonês
CP936	página de código 936, variantes GB2312, GBK ou GB18030 do estilo Microsoft Windows para Chinês Simplificado
CP949	página de código 949, variante EUC-KR ou or Unified Hangul Code de estilo Microsoft Windows para Coreano
CP950	página de código 950, variante Big5 de estilo Microsoft Windows para Chinês Tradicional
CP1251	página de código 1251, codificação estilo Microsoft Windows para o alfabeto Cirílico
CP1252	página de código 1252, variante ISO-8859-15 de estilo Microsoft Windows para linguagens de Europeu ocidental
KOI8-R	antigo standard Russo de UNIX para o alfabeto Cirílico
ISO-2022-JP	codificação standard para email Japonês que usar apenas códigos de 7 bits
eucJP	antigo standard Unix de Japonês de código de 8 bits e completamente diferente do Shift-JIS
Shift-JIS	JIS X 0208 Appendix 1 standard para Japonês (veja CP932)

Tabela 11.2: Lista de valores de codificação e a utilização deles

Nota
Algumas codificações são apenas suportadas para conversão de dados e não são usados como valores do regionais (Secção 8.1).

Para os conjuntos de caracteres que cabem num byte único como os conjuntos de caracteres ASCII e ISO-8859, a codificação de caracteres significa quase o mesmo que o conjunto de caracteres.

Para conjuntos de caracteres com muitos caracteres como o JIS X 0213 para Japonês ou Universal Character Set (UCS, Unicode, ISO-10646-1) para praticamente todas as linguagens, existem muitos esquemas de codificação para os pôr na sequência dos dados do byte.

- EUC e ISO/IEC 2022 (também conhecido como JIS X 0202) para Japonês
- UTF-8, UTF-16/UCS-2 e UTF-32/UCS-4 para Unicode

Para estes, existem diferenciações claras entre o conjunto de caracteres e a codificação de caracteres.

A página de código é usada como o sinónimo para as tabelas de codificação de caracteres para alguns específicos de marcas.

Nota
Por favor note que a maioria dos sistemas de codificação partilham o mesmo código com o ASCII para caracteres de 7 bits. Mas há algumas exceções. Se está a converter programas C antigos Japoneses e dados de URLs a partir do casualmente chamado formato de codificação shift-JIS no formato UTF-8, use "`CP932`" como o nome de codificação em vez de "`shift-JIS`" para obter os resultados esperados: `0x5C` → "\" e `0x7E` → "~". Caso contrário, estes são convertidos para caracteres errados.

Dica
O recode(1) também pode ser usado e oferece mais do que as funcionalidades combinadas do iconv(1), fromdos(1), todos(1), frommac(1) e tomac(1). Para mais, veja "`info recode`".

11.1.2 Verifica ficheiro se é UTF-8 com o iconv

Pode verificar se um ficheiro de texto está codificado em UTF-8 com o iconv(1) com o seguinte.

```
$ iconv -f utf8 -t utf8 input.txt >/dev/null || echo "non-UTF-8 found"
```

Dica
Use a opção "`--verbose`" no exemplo em cima para encontrar o primeiro caractere não-UTF-8.

11.1.3 Converter os nomes dos ficheiros com o iconv

Aqui está um script de exemplo para converter a codificação dos nomes de ficheiros daqueles criados sob sistemas operativos antigos para os modernos de UTF-8 num único diretório.

```
#!/bin/sh
ENCDN=iso-8859-1
for x in *;
 do
 mv "$x" "$(echo "$x" | iconv -f $ENCDN -t utf-8)"
done
```

A variável "`$ENCDN`" especifica a codificação original usada para nomes de ficheiros sob SOs mais antigos em Tabela 11.2.

Para um caso mais complicado, por favor monte um sistema de ficheiros (ex. uma partição de uma unidade de disco) que contenha tais nomes de ficheiros com a codificação apropriada como opção do mount(8) (veja Secção 8.1.3) e copie o conteúdo dele inteiro para outro sistema de ficheiros montado como UTF-8 com o comando "`cp -a`".

11.1.4 conversão EOL

O formato de ficheiro de texto, especificamente o código de fim de linha (EOL), é dependente da plataforma.

plataforma	código EOL	control	decimal	hexadecimal
Debian (unix)	LF	^J	10	0A
MSDOS e Windows	CR-LF	^M^J	13 10	0D 0A
Macintosh da Apple	CR	^M	13	0D

Tabela 11.3: Lista de estilos EOL para diferentes plataformas

Os programas de conversão de formato EOL, fromdos(1), todos(1), frommac(1), e tomac(1), são muito úteis. O recode(1) também é útil.

Nota
Alguns dados no sistema Debian, como os dados da página wiki para o pacote `python-moinmoin`, usam o estilo MSDOS (CR-LF) como o código de EOL. Então a regra em cima é apenas uma regra geral.

Nota
A maioria dos editores (ex. `vim`, `emacs`, `gedit`, ...) podem lidar com ficheiros em estilo EOL de MSDOS transparentemente.

Dica
O uso de "`sed -e '/\r$/!s/$/\r/'`" em vez de todos(1) é melhor quando pretende unificar o estilo de EOL para o estilo do MSDOS a partir da mistura de estilos de MSDOS e Unix. (ex. após fundir 2 ficheiros de estilo MSDOS com o diff3(1).) Isto porque o `todos` adiciona CR a todas as linhas.

11.1.5 Conversão de TAB

Existem alguns programas populares especializados para converter os códigos de tab.

função	`bsdmainutils`	`coreutils`
expande tab para espaços	"`col -x`"	`expand`
contrai tab a partir de espaços	"`col -h`"	`unexpand`

Tabela 11.4: Lista de comandos de conversão de TAB dos pacotes `bsdmainutils` e `coreutils`

indent(1) do pacote `indent` reformata completamente os espaços em branco no programa C.

Os programas editores como o `vim` e o `emacs` também podem ser usados para conversão de TAB. Por exemplo com o `vim`, pode expandir a TAB com a sequência de comandos "`:set expandtab`" e "`:%retab`". Pode reverter isto com a sequência de comandos "`:set noexpandtab`" e "`:%retab!`".

11.1.6 Editores com auto-conversão

Os editores modernos inteligentes como o programa `vim` são bastante inteligentes e lidam bem com quaisquer sistemas de codificação e quaisquer formatos de ficheiro. Deve usar estes editores sob o locale UTF-8 numa consola com capacidades de UTF-8 para melhor compatibilidade.

Um antigo ficheiro de texto Unix em Europeu ocidental, "`u-file.txt`", armazenado com a codificação latin1 (iso-8859-1) pode ser editado com o `vim` com o seguinte.

```
$ vim u-file.txt
```

Isto é possível porque o mecanismo de auto detecção da codificação do ficheiro no `vim` assume primeiro a codificação UTF-8 e, se falhar, assume que é latin1.

Um antigo ficheiro de texto Unix em Polaco, "`pu-file.txt`", armazenado com a codificação latin2 (iso-8859-2) pode ser editado com o `vim` com o seguinte.

```
$ vim '+e ++enc=latin2 pu-file.txt'
```

Um antigo ficheiro de texto unix em Japonês, "`ju-file.txt`", armazenado com a codificação eucJP pode ser editado com o `vim` com o seguinte.

```
$ vim '+e ++enc=eucJP ju-file.txt'
```

Um antigo ficheiro de texto do MS Windows em Japonês, "`jw-file.txt`", armazenado na chamada codificação shift-JIS (mais precisamente: CP932) pode ser editado com o `vim` com o seguinte.

```
$ vim '+e ++enc=CP932 ++ff=dos jw-file.txt'
```

Quando um ficheiro é aberto com as opções "`++enc`" e "`++ff`", o "`:w`" na linha de comandos do Vim guarda-o no formato original e sobrescreve o ficheiro original. Também pode especificar o formato de gravação e o nome do ficheiro na linha de comandos do Vim, ex., "`:w ++enc=utf8 new.txt`".

Por favor consulte o mbyte.txt "suporte a texto multi-byte" na ajuda on-line do `vim` e Tabela 11.2 para os valores de locale usados com "`++enc`".

A família de programas `emacs` pode executar as funções equivalentes.

11.1.7 Extracção de texto simples

O seguinte lê uma página web para um ficheiro de texto. Isto é muito útil quando se copia as configurações da Web ou se aplica ferramentas de texto básicas do Unix como o grep(1) numa página web.

```
$ w3m -dump https://www.remote-site.com/help-info.html >textfile
```

De modo semelhante, pode extrair dados de texto simples a partir de outros formatos a usar o seguinte.

11.1.8 Destacar e formatar dados de texto simples

Pode destacar e formatar dados de texto simples com o seguinte.

11.2 Dados XML

A The Extensible Markup Language (XML) é uma linguagem de marcação para documentos que contêm informação estruturada.

Veja informação de introdução em XML.COM.

- "O que é XML?"
- "O que é XSLT?"
- "O que é XSL-FO?"
- "O que é XLink?"

pacote	popcon	tamanho	palavra chave	função
w3m	V:10, I:129	2853	html→text	Conversor de HTML para texto com o comando "w3m -dump"
html2text	V:3, I:66	298	html→text	Conversor de HTML para texto avançado (ISO 8859-1)
lynx	V:29, I:455	2031	html→text	Conversor de HTML para texto com o comando "lynx -dump"
elinks	V:2, I:15	1791	html→text	Conversor de HTML para texto com o comando "elinks -dump"
links	V:2, I:20	2321	html→text	Conversor de HTML para texto com o comando "links -dump"
links2	V:0.8, I:9.4	5466	html→text	Conversor de HTML para texto com o comando "links2 -dump"
catdoc	V:16, I:170	682	MSWord→text,tex	converte ficheiros do MSWord para texto simples ou TeX
antiword	V:0.9, I:6.3	596	MSWord→text,ps	converte ficheiros do MSWord para texto simples ou ps
unhtml	V:0.04, I:0.49	40	html→text	remove as etiquetas de marcas de um ficheiro HTML
odt2txt	V:1, I:19	60	odt→text	conversor de texto do OpenDocument para texto

Tabela 11.5: Lista de ferramentas para extracção de dados de texto simples

pacote	popcon	tamanho	palavra chave	descrição
vim-runtime	V:17, I:355	38849	destaque	MACRO do Vim para converter código fonte em HTML com ":source $VIMRUNTIME/syntax/html.vim"
cxref	V:0.01, I:0.21	1191	c→html	conversor de programa C para latex e HTML (linguagem C)
src2tex	V:0.02, I:0.16	1799	destaque	converte muitos códigos fonte para TeX (linguagem C)
source-highlight	V:0.4, I:2.9	2131	destaque	converte muitos códigos fonte para HTML, XHTML, LaTeX, Texinfo, sequências de escape do cores ANSI e ficheiros do DocBook com destaques (C++)
highlight	V:0.4, I:2.8	1412	destaque	converte muitos códigos fonte para HTML, XHTML, RTF, LaTeX, TeX ou ficheiros XSL-FO com destaques (C++)
grc	V:0.9, I:5.7	208	text→color	colorizador genérico para tudo (Python)
pandoc	V:11, I:47	209559	text→any	conversor geral de markup (Haskell)
python3-docutils	V:13, I:51	2013	text→any	Formatador de documento de Texto Re-Estruturado para XML (Python)
markdown	V:0.4, I:5.7	56	text→html	Formatador de documentos de texto Markdown para (X)HTML (Perl)
asciidoctor	V:0.5, I:4.6	101	text→any	Formatador de documentos de texto AsciiDoc para XML/HTML (Ruby)
python3-sphinx	V:7, I:27	3234	text→any	Sistema de publicação de documentos baseado em texto reestruturado (Python)
hugo	V:0.8, I:5.0	66708	text→html	Sistema de publicação de sites estáticos baseado em Markdown (Go)

Tabela 11.6: Lista de ferramentas para destacar dados em texto simples

11.2.1 Dicas básicas para XML

O texto em XML parece-se com HTML. Permite-nos gerir múltiplos formatos de saída de um documento. Um sistema XML fácil é o pacote `docbook-xsl`, o qual é usado aqui.

Cada ficheiro XML começa com a declaração XML standard como o seguinte.

```
<?xml version="1.0" encoding="UTF-8"?>
```

A sintaxe básica para um elemento XML é marcado como a seguir.

```
<name attribute="value">content</name>
```

O elemento XML com conteúdo vazio é marcado no seguinte formato curto.

```
<name attribute="value" />
```

O "`atributo="valor"`" nos exemplos em cima é opcional.

A secção de comentários em XML está marcada como a seguir.

```
<!-- comment -->
```

Em vez de adicionar marcações, o XML requer conversão menor ao conteúdo a usar entidades predefinidas para os seguintes caracteres.

entidade predefinida	**caractere a ser convertido em**
`"`	" : cotação
`'`	' : apóstrofo
`<`	< : menor-que
`>`	> : maior-que
`&`	& : E comercial

Tabela 11.7: Lista de entidades predefinidas para XML



Cuidado
"<" ou "&" não podem ser usados em atributos ou elementos.

Nota
Quando são utilizadas entidades definidas pelo utilizador ao estilo SGML, por exemplo,"`&some-tag;`", a primeira definição ganha às outras. A definição da entidade é expressa em "`<!ENTITY some-tag "entity value">`".

Nota
Desde que as marcações de XML sejam feitas de modo consistente com um certo conjunto de nomes de etiquetas (em vez de alguns dados como conteúdo ou valor de atributo), a conversão para outro XML é uma tarefa trivial a usar Extensible Stylesheet Language Transformations (XSLT).

11.2.2 Processamento de XML

Existem muitas ferramentas disponíveis para processar ficheiros XML como o Extensible Stylesheet Language (XSL).

Basicamente, após criar um ficheiro XML bem formado, pode convertê-lo para qualquer formato a usar o Extensible Stylesheet Language Transformations (XSLT).

A Extensible Stylesheet Language for Formatting Objects (XSL-FO) é suposto ser a solução para a formatação. O pacote `fop` é novo no arquivo `main` de Debian devido à sua dependência da linguagem de programação Java. Por isso o código LaTeX é normalmente gerado a partir de XML a utilizar XSLT e o sistema LaTeX é utilizado para criar ficheiros imprimíveis tais como DVI, PostScript e PDF.

pacote	popcon	tamanho	palavra chave	descrição
`docbook-xml`	V:15, I:407	2126	xml	definição de tipo de documento XML (DTD) para DocBook
`docbook-xsl`	V:15, I:144	14823	xml/xslt	folhas de estilo XSL para processar XML do DocBook para vários formatos de saída com XSLT
`xsltproc`	V:15, I:71	83	xslt	processador de linha de comandos XSLT (XML→ XML, HTML, texto simples, etc.)
`xmlto`	V:0.5, I:8.0	124	xml/xslt	conversor de XML-para-qualquer com XSLT
`fop`	V:0.7, I:8.0	281	xml/xsl-fo	converter ficheiros Docbook XML para PDF
`dblatex`	V:0.9, I:5.4	4635	xml/xslt	converte ficheiros do Docbook para documentos DVI, PostScript, PDF com o XSLT
`dbtoepub`	V:0.04, I:0.54	37	xml/xslt	conversor de XML DocBook para .epub

Tabela 11.8: Lista de ferramentas XML

Como o XML é um sub-conjunto da Standard Generalized Markup Language (SGML), pode ser processado pelas ferramentas extensivas disponíveis para SGML, como o Document Style Semantics and Specification Language (DSSSL).

pacote	popcon	tamanho	palavra chave	descrição
`openjade`	V:1, I:21	1066	`dsssl`	ISO/IEC 10179:1996 processador DSSSL standard (mais recente)
`docbook-dsssl`	V:0.5, I:7.4	2594	`xml/dsssl`	folhas de estilo DSSSL para processar XML do DocBook para vários formatos de saída com DSSSL
`docbook-utils`	V:0.3, I:5.2	287	`xml/dsssl`	utilitários para ficheiros do DocBook incluindo a conversão para outros formatos (HTML, RTF, PS, man, PDF) com comandos `docbook2*` com DSSSL

Tabela 11.9: Lista de ferramentas DSSSL

Dica
No GNOME o `yelp` é por vezes, útil para ler ficheiros DocBook XML diretamente, uma vez que é renderizado decentemente no X.

11.2.3 A extracção de dados de XML

Pode extrair dados de HTML ou XML a partir de outros formatos a usar os seguintes.

pacote	popcon	tamanho	palavra chave	descrição
man2html	V:0.1, I:1.2	142	manpage→html	conversor de manual (manpage) para HTML (suporte a CGI)
doclifter	V:0.00, I:0.05	487	troff→xml	conversor de troff para DocBook XML
texi2html	V:0.2, I:2.8	1847	texi→html	conversor de Texinfo para HTML
info2www	V:0.8, I:1.4	76	info→html	conversor de info do GNU para HTML (suporte a CGI)
wv	V:0.2, I:2.3	733	MSWord→any	conversor de documentos Microsoft Word para HTML, LaTeX, etc.
unrtf	V:0.3, I:2.8	159	rtf→html	conversor de documentos de RTF para HTML, etc
wp2x	V:0.00, I:0.08	200	WordPerfect→any	Ficheiros do WordPerfect 5.0 e 5.1 para TeX, LaTeX, troff, GML e HTML

Tabela 11.10: Lista de ferramentas de extracção de dados de XML

11.2.4 O lint de dados XML

Para ficheiros HTML não-XML, pode convertê-los para XHTML o que é uma instância de XML bem formado. O XHTML pode ser processado por ferramentas de XML.

É possível verificar a sintaxe dos ficheiros XML e a qualidade dos URLs neles contidos.

pacote	popcon	tamanho	função	descrição
libxml2-utils	V:60, I:207	213	xml↔html↔xhtml	ferramenta de XML de linha de comandos com xmllint(1) (verificação de sintaxe, reformatação, lint, …)
tidy	V:0.9, I:6.9	79	xml↔html↔xhtml	Verificador e reformatador de sintaxe HTML
weblint-perl	V:0.05, I:0.90	32	lint	Verificado de sintaxe e estilo mínimo para HTML
linklint	V:0.04, I:0.44	343	verificação da ligação	verificador de ligações rápido e ferramenta de manutenção de sites web

Tabela 11.11: Lista de ferramentas de impressão bonita de XML

Após o XML apropriado ser gerado, pode usar a tecnologia XSLT para extrair dados baseados no contexto de marcações e etc.

11.3 Formatação de texto

O programa troff do Unix desenvolvido originalmente pela AT&T pode ser usado para formatação de texto simples. É geralmente usado para criar as páginas de manual (manpages).

O TeX criado por Donald Knuth é uma ferramenta de formatação de texto muito poderosa e é o standard de facto. O LaTeX originalmente escrito por Leslie Lamport permite um acesso de alto nível ao poder do TeX.

11.3.1 formatação de texto roff

Tradicionalmente, o roff é o sistema de processamento de texto principal 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`".

pacote	popcon	tamanho	palavra chave	descrição
texlive	V:1, I:27	55	(La)TeX	sistema TeX para formatação de texto, pre-visualização e impressão
lilypond	V:0.7, I:5.4	16924	(La)TeX	Máquina de compor música
groff	V:2, I:23	16536	troff	O sistema de formato de texto troff do GNU

Tabela 11.12: Lista de ferramentas de formatação de texto

Pode ler ou imprimir um bom tutorial e referência em "-me" macro em "/usr/share/doc/groff/" instalando o pacote groff.

Dica
"groff -Tascii -me -" produz resultados em texto simples com código de escape ANSI. Se deseja obter resultados tipo manpage com muitos "^H" e "_", então use "GROFF_NO_SGR=1 groff -Tascii -me -".

Dica
Para remover "^H" e "_" de um ficheiro de texto gerado pelo groff, filtre-o com "col -b -x".

11.3.2 TeX/LaTeX

A distribuição de software TeX Live oferece um sistema TeX completo. O meta-pacote texlive disponibiliza uma seleção decente dos pacotes TeX Live que deverão ser suficientes para as tarefas mais comuns.

Existem muitas referências disponíveis para TeX e LaTeX.

- O HOWTO do The teTeX: O Guia Local de Linux-teTeX
- tex(1)
- latex(1)
- texdoc(1)
- texdoctk(1)
- "The TeXbook", por Donald E. Knuth, (Addison-Wesley)
- "LaTeX - A Document Preparation System", por Leslie Lamport, (Addison-Wesley)
- "The LaTeX Companion", por Goossens, Mittelbach, Samarin, (Addison-Wesley)

Este é o ambiente de formatação de texto mais poderoso. Muitos processadores SGML usam isto como processador de texto em backend. O Lyx disponibilizado pelo pacote lyx e o GNU TeXmacs disponibilizado pelo pacote texmacs oferecem um bom ambiente de edição OQVEOQT para o LaTeX enquanto muitos usam o Emacs e o Vim como a sua escolha para editor de código fonte.

Existem muitos recursos online disponíveis.

- O Guia TEX Live - TEX Live 2007 ("/usr/share/doc/texlive-doc-base/english/texlive-en/live.html") (pacote texlive-doc-base package)
- Um Guia Simples para o Latex/Lyx
- Processamento de Texto a Usar o LaTeX

Quando os documentos ficam maiores, por vezes o TeX pode causar erros. tem de aumentar o tamanho do pool em "`/etc/texmf/texmf.cnf`" (ou mais apropriadamente editar o "`/etc/texmf/texmf.d/95NonPath`" e correr update-texmf(8)) para corrigir isto.

Nota
A fonte TeX de "The TeXbook" está disponível em www.ctan.org site de arquivo-tex para texbook.tex. Este ficheiro contém a maior parte das macros necessárias. Ouvi dizer que pode processar este documento com o tex(1) depois de comentar as linhas 7 a 10 e adicionar "`\input manmac \proofmodefalse`". Recomenda-se fortemente a compra deste livro (e de todos os outros livros de Donald E. Knuth) em vez de usar a versão online, mas a fonte é um ótimo exemplo de entrada TeX!

11.3.3 Impressão bonita de um manual

Consegue uma impressão bonita dum manual em PostScript com um dos seguintes comandos.

```
$ man -Tps some_manpage | lpr
```

11.3.4 Criar um manual

Apesar de escrever um manual (manpage) no formato troff simples ser possível, existem alguns programas que ajudam a criá-lo.

pacote	popcon	tamanho	palavra chave	descrição
`docbook-to-man`	V:0.6, I:5.3	189	`SGML`→`manpage`	conversor de DocBook SGML para macros roff man
`help2man`	V:0.5, I:6.1	542	`text`→`manpage`	geração automática de manual a partir do --help
`info2man`	V:0.02, I:0.18	134	`info`→`manpage`	conversor de info do GNU para POD ou páginas man
`txt2man`	V:0.04, I:0.61	112	`text`→`manpage`	converte texto ASCII simples para o formato de página man

Tabela 11.13: Lista de pacotes para ajudar a criar o manual (manpage)

11.4 Dados imprimíveis

Os dados imprimíveis são expressos no formato PostScript no sistema Debian. O Common Unix Printing System (CUPS) usa o Ghostscript como o programa backend de rasterização dele para as impressoras não-PostScript.

Os dados imprimíveis também podem ser expressos no formato PDF no recente sistema Debian.

Os ficheiros PDF podem ser visualizados e as suas entradas de formulário podem ser preenchidas utilizando ferramentas de visualização GUI como o Evince e o Okular (ver Secção 7.4); e navegadores modernos como o Chromium.

Os ficheiros PDF podem ser editados utilizando algumas ferramentas gráficas como o LibreOffice, o Scribus e o Inkscape (ver Secção 11.6).

Dica
É possível ler um ficheiro PDF com o GIMP e convertê-lo para o formato PNG com uma resolução superior a 300 dpi. Esta pode ser utilizada como imagem de fundo para o LibreOffice para produzir uma impressão alterada desejável com o mínimo de esforço.

11.4.1 Ghostscript

O núcleo da manipulação de dados a imprimir é o interpretador Ghostscript PostScript (PS) o qual gera imagem rasterizada.

pacote	popcon	tamanho	descrição
ghostscript	V:165, I:559	177	O interpretador de PostScript/PDF Ghostscript GPL
ghostscript-x	V:0, I:13	88	Interpretador de PostScript/PDF Ghostscript GPL - suporte a ecrã X
libpoppler162	V:14, I:22	5313	Biblioteca de renderização de PDF que é um fork do visualizador de PDF xpdf
libpoppler-glib8t64	V:77, I:323	612	Biblioteca de renderização de PDF (biblioteca de partilha baseada em GLib)
poppler-data	V:161, I:582	13086	CMaps para suporte à biblioteca de renderização de PDF (para CJK: Adobe-*)

Tabela 11.14: Lista de interpretadores PostScript Ghostscript

Dica
"`gs -h`" pode mostrar a configuração do Ghostscript.

11.4.2 Juntar dois ficheiros PS ou PDF

Pode unir dois ficheiros PostScript (PS) ou Portable Document Format (PDF) a usar o gs(1) do Ghostscript.

```
$ 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
```

Nota
O PDF, que é um formato de dados imprimíveis amplamente usado em várias plataformas, é essencialmente o formato PS comprimido com algumas funcionalidades e extensões adicionais.

Dica
Para a linha de comandos, o psmerge(1) e outros comandos do pacote `psutils` são úteis para manipular documentos em PostScript. O pdftk(1) do pacote `pdftk` também é útil para manipular documentos em PDF.

11.4.3 Utilitários de dados imprimíveis

Os seguintes pacotes para utilitários de dados imprimíveis chamaram a minha atenção.

11.4.4 Imprimir com o CUPS

Ambos comandos lp(1) e lpr(1) oferecidos pelo Common Unix Printing System (CUPS) disponibilizam opções para impressão personalizada dos dados a imprimir.

Pode imprimir 3 cópias coligidas de um ficheiro a usar um dos seguintes comandos.

```
$ lp -n 3 -o Collate=True filename
```

pacote	popcon	tamanho	palavra chave	descrição
poppler-utils	V:169, I:476	804	pdf→ps,text	Utilitários de PDF: pdftops, pdfinfo, pdfimages, pdftotext, pdffonts
psutils	V:4, I:48	34	ps→ps	Ferramentas de conversão de documentos PostScript
poster	V:0.1, I:1.6	57	ps→ps	criar grandes posters de páginas PostScript
enscript	V:1, I:10	2138	text→ps, html, rtf	converter texto ASCII para PostScript, HTML, RTF ou Pretty-Print
a2ps	V:0.7, I:6.7	4109	text→ps	Conversor de 'Qualquer coisa para PostScript' e pretty-printer
pdftk	V:1, I:20	28	pdf→pdf	Ferramenta de conversão de documentos PDF: pdftk
html2ps	V:0.2, I:1.7	256	html→ps	conversor de HTML para PostScript
gnuhtml2latex	V:0.03, I:0.57	26	html→latex	conversor de html para latex
latex2rtf	V:0.1, I:1.9	495	latex→rtf	conversor de documentos LaTeX para RTF que podem ser lidos pelo MS Word
ps2eps	V:2, I:31	95	ps→eps	conversor de PostScript para EPS (PostScript Encapsulado)
e2ps	V:0.01, I:0.11	104	text→ps	Conversor de texto para PostScript com suporte a codificação Japonesa
impose+	V:0.1, I:1.6	118	ps→ps	Utilitários do PostScript
trueprint	V:0.00, I:0.07	148	text→ps	impressão bonita de muitos códigos fonte (C, C++, Java, Pascal, Perl, Pike, Sh e Verilog) para PostScript. (linguagem C)
pdf2svg	V:0.2, I:2.9	33	pdf→svg	conversor de PDF para formato Gráficos vectoriais escaláveis
pdftoipe	V:0.01, I:0.43	69	pdf→ipe	conversor de PDF para formato XML de IPE

Tabela 11.15: Lista de utilitários de dados imprimíveis

```
$ lpr -#3 -o Collate=True filename
```

Pode personalizar ainda mais as operações da impressora ao usar opções da impressão como "`-o number-up=2`", "`-o page-set=even`", "`-o page-set=odd`", "`-o scaling=200`", "`-o natural-scaling=200`", etc., documentadas em Impressão em Linha de Comandos e Opções.

11.5 A conversão de dados de mail

Os seguintes pacotes para conversão de dados de mail chamaram a minha atenção.

pacote	popcon	tamanho	palavra chave	descrição
`sharutils`	V:2, I:27	1436	mail	shar(1), unshar(1), uuencode(1), uudecode(1)
`mpack`	V:0.7, I:7.4	109	`MIME`	codificação e descodificação de mensagens MIME: mpack(1) e munpack(1)
`tnef`	V:0.3, I:3.7	103	`ms-tnef`	descompactar anexos MIME do tipo "application/ms-tnef" o qual é um formato apenas da Microsoft
`uudeview`	V:0.2, I:1.7	105	`mail`	codificador e descodificador dos seguintes formatos: uuencode, xxencode, BASE64, quoted printable e BinHex

Tabela 11.16: Lista de pacotes para ajudar na conversão de dados de mail

Dica
O servidor IMAP4 (Internet Message Access Protocol versão 4) pode ser utilizado para transferir correio eletrónico de sistemas de correio proprietários se o software cliente de correio puder ser configurado para utilizar também o servidor IMAP4.

11.5.1 Noções básicas de dados de mail

Os dados do correio eletrónico (SMTP) devem limitar-se a séries de dados de 7 bits. Assim, os dados binários e os dados de texto de 8 bits são codificados em formato de 7 bits com as Extensões de Correio da Internet para Fins Múltiplos (MIME) e a seleção do conjunto de caracteres (ver Tabela 11.2).

O formato de armazenamento de mail standard é mbox de acordo com RFC2822 (RFC822 atualizado). Veja mbox(5) (disponibilizado pelo pacote `mutt`).

Para as linguagens Europeias, "`Content-Transfer-Encoding: quoted-printable`" com o junto de caracteres ISO-8859-1 é geralmente usado para mail porque não existem muitos caracteres de 8 bits. Se o texto Europeu estiver codificado em UTF-8, é provável que seja usado o "`Content-Transfer-Encoding: quoted-printable`" pois é maioritariamente dados de 7 bits.

Para Japonês, tradicionalmente usa-se "`Content-Type: text/plain; charset=ISO-2022-JP`" para o mail para manter o texto em 7 bits. Mas os sistemas mais antigos da Microsoft podem enviar dados de mail em Shift-JIS sem a declaração apropriada. Se o testo Japonês for codificado em UTF-8, é provável que se use Base64 pois contém muitos dados de 8 bits. A situação de outras linguagens Asiáticas é semelhante.

Nota
Se os seus dados de correio não-Unix são acessíveis por um software cliente não-Debian que pode falar com o servidor IMAP4, pode ser capaz de os mover executando o seu próprio servidor IMAP4.

Nota
Se usa outros formatos de armazenamento de mail, movê-los para o formato mbox é um bom primeiro passo. Um programa cliente versátil como o mutt(1) pode dar jeito para isto.

Pode dividir os conteúdos da mailbox para cada mensagem a usar o procmail(1) e o formail(1).

Cada mensagem de mail pode ser desempacotada a usar o munpack(1) do pacote `mpack` (ou outras ferramentas especializadas) para obter os conteúdos codificados em MIME.

11.6 Ferramentas de dados gráficos

Apesar dos programas com GUI como o gimp(1) serem muito poderosos, as ferramentas de linha de comandos como o imagemagick(1) são bastante úteis para automatizar a manipulação de imagens via scripts.

O formato de facto de ficheiros de imagem das câmaras digitais é o Exchangeable Image File Format (EXIF) o qual é o formato de ficheiros de imagem JPEG com etiquetas de meta-dados adicionais. Pode conter informações como a data, hora e definições da camera.

A patente de compressão de dados sem perdas Lempel-Ziv-Welch (LZW) expirou. Os utilitários de Graphics Interchange Format (GIF) que usam o método de compressão LZW estão agora disponíveis livremente no sistema Debian.

Dica
Qualquer câmara digital ou scanner com meio de gravação amovível funciona em Linux através de leitores de armazenamento USB desde que sigam as Regras de Desenho para Sistemas de Ficheiros de Câmaras e usem o sistema de ficheiros FAT. Veja Secção 10.1.7.

11.6.1 Ferramentas de dados gráficos (meta-pacote)

Os seguintes meta-pacotes são bons pontos de partida para procurar ferramentas de dados gráficos usando o aptitude(8). "Vista geral de pacotes para Maintainers Debian de Ferramentas de Fotografia" pode ser outro ponto de partida.

pacote	popcon	tamanho	palavra chave	descrição
`education-graphics`	I:0.34	26	`svg,jpeg,…`	meta-pacote para o ensino do grafismo e da arte pictórica.
`open-font-design-toolkit`	I:0.04	9	`ttf,ps,…`	meta pacote para desenho de font aberta

Tabela 11.17: Lista de ferramentas de dados gráficos (meta-pacote)

Dica
Procure mais ferramentas de imagem com a expressão regular "`~Gworks-with::image`" no aptitude(8) (veja Secção 2.2.6).

11.6.2 Ferramentas gráficas de dados (GUI)

Os seguintes pacotes para as ferramentas de conversão, edição e organização de dados gráficos GUI chamaram a minha atenção.

pacote	popcon	tamanho	palavra chave	descrição
gimp	V:34, I:211	33315	image(bitmap)	GNU Image Manipulation Program - GIMP
xsane	V:9, I:128	1512	image(bitmap)	Frontend X11 baseado em GTK para o SANE (Scanner Access Now Easy)
scribus	V:1, I:12	32942	ps/pdf/SVG/...	editor de DTP do Scribus
libreoffice-draw	V:89, I:418	11418	image(vector)	suite de escritório do LibreOffice - desenho
inkscape	V:12, I:76	113353	image(vector)	editor de SVG (Scalable Vector Graphics)
dia	V:1, I:16	3846	image(vector)	editor de diagramas (Gtk)
xfig	V:0.5, I:8.5	7951	image(vector)	Habilidade para Geração interactiva de figuras sob X11
gocr	V:0.5, I:3.6	549	image→text	software de OCR livre
eog	V:23, I:129	10535	image(Exif)	Programa de visualização de gráficos Olho do GNOME
gthumb	V:3, I:11	5162	image(Exif)	visualizador e navegador de imagens (GNOME)
geeqie	V:3, I:11	3367	image(Exif)	Visualizador de imagens que usa GTK
shotwell	V:14, I:246	6334	image(Exif)	organizador de fotos digitais (GNOME)
gwenview	V:41, I:114	6057	image(Exif)	visualizador de imagens (KDE)
kamera	I:114	992	image(Exif)	Suporte para cameras digitais para aplicações do KDE
digikam	V:1.5, I:8.3	333	image(Exif)	aplicação de gestão de fotos digitais para KDE
darktable	V:4, I:11	36092	image(Exif)	mesa de luz virtual e câmara escura para fotógrafos
hugin	V:0.5, I:5.3	6524	image(Exif)	fotomontagem de fotografias panorâmicas
librecad	V:1, I:13	9164	DXF,…	Editor de dados CAD 2D
freecad	V:1, I:20	112	DXF,…	Editor de dados CAD 3D
blender	V:2, I:20	171893	blend,TIFF,VRML,…	editor de conteúdos 3D para animação e etc
mm3d	V:0.02, I:0.22	4123	ms3d,obj,dxf,…	editor de modelos 3D baseado em OpenGL
fontforge	V:0.6, I:5.3	4736	ttf,ps,…	editor de tipo de letra para fonts PS, TrueType e OpenType
xgridfit	V:0.02, I:0.08	878	ttf	programa para ajustes e alinhamento em grelha de tipos de letra TrueType

Tabela 11.18: Lista de ferramentas de dados gráficos (GUI)

11.6.3 Ferramentas de dados gráficos (CLI)

Os seguintes pacotes para as ferramentas de conversão, edição e organização de dados gráficos CLI chamaram a minha atenção.

11.7 Conversão de dados variados

Existem muitos outros programas para converter dados. Os pacotes seguintes chamaram a minha atenção a usar a expressão regular "`~Guse::converting`" no aptitude(8) (veja Secção 2.2.6).

Também pode extrair dados do formato RPM com o seguinte.

```
$ rpm2cpio file.src.rpm | cpio --extract
```

pacote	popcon	tamanho	palavra chave	descrição
imagemagick	V:7, I:272	80	image(bitmap)	programas de manipulação de imagens
graphicsmagick	V:1.0, I:8.3	5945	image(bitmap)	programas de manipulação de imagens (fork do imagemagick)
netpbm	V:27, I:282	8435	image(bitmap)	ferramentas de conversão de gráficos
libheif-examples	V:0.3, I:3.6	528	heif→jpeg(bitmap)	converter o formato de ficheiro de imagem de alta eficiência (HEIF) para os formatos JPEG, PNG ou Y4M com o comando heif-convert(1)
icoutils	V:3, I:33	221	png↔ico(bitmap)	converte ícones e cursores do MS Windows de e para formatos PNG (favicon.ico)
pstoedit	V:2, I:37	1075	ps/pdf→image(vector)	conversor de ficheiro PostScript e PDF para gráficos vectoriais editáveis (SVG)
libwmf-bin	V:5, I:80	149	Windows/imagem(vector)	ferramentas de conversão de meta-ficheiros do Windows (dados de gráficos vectoriais)
fig2sxd	V:0.01, I:0.16	158	fig→sxd(vector)	converte ficheiros XFig ao formato do OpenOffice.org Draw
unpaper	V:2, I:16	417	image→image	ferramenta de pós-processamento para páginas digitalizadas em scanner para OCR
tesseract-ocr	V:8, I:34	2209	image→text	software livre de OCR baseado no motor de OCR comercial da HP
tesseract-ocr-eng	V:9, I:35	4032	image→text	Dados de motor OCR: ficheiros de linguagem tesseract-ocr para texto Inglês
ocrad	V:0.3, I:2.2	608	image→text	software de OCR livre
exif	V:3, I:49	335	image(Exif)	utilitário de linha de comandos para mostrar informação EXIF nos ficheiros JPEG
exiv2	V:1, I:18	436	image(Exif)	ferramenta de manipulação de meta-dados EXIF/IPTC
exiftran	V:1, I:11	81	image(Exif)	transformar imagens jpeg de câmaras digitais
exiftags	V:0.2, I:2.5	309	image(Exif)	utilitário para ler etiquetas Exif de ficheiros JPEG de câmaras digitais
exifprobe	V:0.2, I:2.1	506	image(Exif)	ler meta-dados de imagens digitais
dcraw	V:0.7, I:7.2	428	image(Raw)→ppm	decodifica imagens cruas de câmaras digitais
findimagedupes	V:0.11, I:0.96	76	image→fingerprint	encontra imagens visualmente semelhantes ou duplicadas
ale	V:0.00, I:0.14	850	image→image	junta imagens para aumentar a fidelidade ou criar mosaicos
imageindex	V:0.1, I:1.1	143	image(Exif)→html	cria galerias HTML estáticas a partir de imagens
outguess	V:0.1, I:1.0	230	jpeg,png	ferramenta de Esteganografia universal
jpegoptim	V:0.6, I:5.8	59	jpeg	otimize ficheiros JPEG
optipng	V:2, I:39	187	png	otimize ficheiros PNG, compressão sem perdas
pngquant	V:1, I:11	62	png	otimize ficheiros PNG, compressão com perdas

Tabela 11.19: Lista de ferramentas de dados gráficos (CLI)

pacote	popcon	tamanho	palavra chave	descrição
alien	V:1, I:12	150	rpm/tgz→deb	conversor para pacotes alienígenas num pacote Debian
freepwing	V:0.00, I:0.01	447	EB→EPWING	conversor de ”Electric Book” (popular no Japão) para um formato JIS X 4081 único (um subconjunto de EPWING V1)
calibre	V:7, I:24	71216	any→EPUB	conversor de e-books e gestor de biblioteca

Tabela 11.20: Lista de ferramentas de conversão de dados variados

Capítulo 12

Programação

Disponibilizo algumas dicas para as pessoas aprenderem programação no sistema Debian o suficiente para rastrear o código fonte do pacote. Aqui estão pacotes notáveis e pacotes de documentação correspondentes para programação.

Estão disponíveis referências online ao escrever "`man nome`" após instalar os pacotes `manpages` e `manpages-dev`. As referências online ás ferramentas GNU está disponíveis ao escrever "`info nome_do_programa`" após instalar os pacotes de documentação pertinentes. Poderá ter de incluir os arquivos `contrib` e `non-free` adicionalmente ao arquivo `main` pois algumas documentações GFDL não são consideradas compatíveis com DFSG.

Considere a possibilidade de utilizar ferramentas do sistema de controlo de versões. Ver Secção 10.5.



Atenção
Não use "`test`" como o nome de um ficheiro de teste executável. "`test`" é um comando embutido na shell.



Cuidado
Deve instalar os programas compilados directamente a partir da fonte em "`/usr/local`" ou "`/opt`" para evitar colisões com os programas do sistema.

Dica
Os Exemplos de código da criação de "Song 99 Bottles of Beer" devem dar-lhe uma boa ideia de praticamente todas as linguagens de programação.

12.1 O script de shell

O script de shell é um ficheiro de texto com o bit de execução definido e contém os comandos no seguinte formato.

```
#!/bin/sh
 ... command lines
```

A primeira linha especifica o interpretador shell que lê e executa o conteúdo deste ficheiro.

Ler scripts de shell é a **melhor** maneira de compreender como um sistema tipo Unix funciona. Aqui, Dou alguns apontamentos e lembranças para programação de shell. Veja "Erros de Shell" (https://www.greenend.org.uk/rjk/-2001/04/shell.html) para aprender a partir de erros.

Ao contrário do modo interativo de shell (veja Secção 1.5 e Secção 1.6), os scripts de shell usam frequentemente parâmetros, condicionais e ciclos.

12.1.1 Compatibilidade da shell do POSIX

Muitos scripts de sistema podem ser interpretados por qualquer uma das shells POSIX (ver Tabela 1.13).

- A shell POSIX não-interactiva predefinida "`/usr/bin/sh`" é uma ligação simbólica que aponta para `/usr/bin/dash` e é utilizada por muitos programas de sistema.
- A shell POSIX interactiva predefinida é `/usr/bin/bash`.

Evite escrever um script de shell com **bashisms** ou **zshisms** para fazê-loportável entre todas as shells do POSIX. Pode verificar isto a usar o checkbashisms(1).

Bom: POSIX	**Evitar: 'bashism'**
`if [ "$foo" = "$bar" ] ; then …`	`if [ "$foo" == "$bar" ] ; then …`
`diff -u file.c.orig file.c`	`diff -u file.c{.orig,}`
`mkdir /foobar /foobaz`	`mkdir /foo{bar,baz}`
`funcname() { … }`	`function funcname() { … }`
formato octal: "`\377`"	formato hexadecimal: "`\xff`"

Tabela 12.1: Lista dos 'bashisms' típicos

O comando "`echo`" tem de ser usado com os seguintes cuidados porque a implementação dele difere entre o integrado na shell e os comandos externos.

- Evite usar quaisquer opções de comando excepto "`-n`".
- Evite usar sequências de escape na cadeia porque o manuseamento dele varia.

Nota
Apesar da opção "`-n`" **não** ser realmente sintaxe POSIX, geralmente é aceite.

Dica
Use o comando "`printf`" em vez do comando "`echo`" se precisar de embeber sequências de escape na cadeia de saída.

12.1.2 Parâmetros da shell

Parâmetros de shell especiais são frequentemente usados no script shell.

As **expansões de parâmetro** básicas a lembrar são as seguintes.

Aqui, o símbolo ortográfico dois pontos "`:`" em todas estas operações é na realidade opcional.

- **com** "`:`" = teste de operador para **existe** e **não nulo**
- **sem** "`:`" = teste de operador para apenas **existe**

parâmetro da shell	valor
`$0`	nome da shell ou script de shell
`$1`	primeiro(1) argumento shell
`$9`	nono(9) argumento shell
`$#`	quantidade de parâmetros de posição
`"$*"`	`"$1 $2 $3 $4 … "`
`"$@"`	`"$1" "$2" "$3" "$4" …`
`$?`	estado de saída do comando mais recente
`$$`	PID deste script shell
`$!`	PID da tarefa de fundo iniciada mais recentemente

Tabela 12.2: Lista de parâmetros da shell

formato da expressão do parâmetro	valor se `var` estiver definido	valor se `var` não estiver definido
`${var:-string}`	”`$var`”	”`string`”
`${var:+string}`	”`string`”	”`null`”
`${var:=string}`	”`$var`”	”`sequência de caracteres`” (e corra ”`var=sequência de caracteres`”)
`${var:?string}`	”`$var`”	echo ”`sequência de caracteres`” para **stderr** (e termina com erro)

Tabela 12.3: Lista de expansões de parâmetros de shell

formato de substituição de parâmetro	resultado
`${var%suffix}`	remover o modelo de sufixo menor
`${var%%suffix}`	remover o modelo de sufixo maior
`${var#prefix}`	remover o modelo de prefixo menor
`${var##prefix}`	remover o modelo de prefixo maior

Tabela 12.4: Lista de substituições de parâmetros de shell chave

12.1.3 Condicionais da shell

Cada comando retorna um **estado de saída** que pode ser usado para expressões condicionais.

- Sucesso: 0 (”True”)
- Erro: não 0 (”False”)

Nota
”0” no contexto condicional da shell significa ”Verdadeiro”, enquanto ”0” no contexto condicional de C significa ”Falso”.

Nota
”`[`” é o equivalente do comando `test`, o qual avalia os seus argumentos até ao ”`]`” como uma expressão condicional.

Os **idiomas condicionais** básicos a lembrar são os seguintes.

- ”*`comando`* `&&` *`se_sucesso_corre_também_este_comando`* `||` `true`”
- ”*`comando`* `||` *`se_não_sucesso_corre_também_este_comando`* `||` `true`”
- Um fragmento de script de multi-linhas como o seguinte

```
if [ conditional_expression ]; then
 if_success_run_this_command
else
 if_not_success_run_this_command
fi
```

Aqui o ”`|| true`” final foi necessário para assegurar que estes script de shell não termina acidentalmente nesta linha quando a shell é invocada com a flag ”`-e`”.

equação	condição para retornar o verdadeiro lógico
`-e` *`file`*	*ficheiro* existe
`-d` *`file`*	*ficheiro* existe e é um diretório
`-f` *`file`*	*ficheiro* existe e é um ficheiro normal
`-w` *`file`*	*ficheiro* existe e pode-se escrever nele
`-x` *`file`*	*ficheiro* existe e é executável
`file1` `-nt` *`file2`*	*ficheiro1* é mais recente que *ficheiro2* (modificação)
`file1` `-ot` *`file2`*	*ficheiro1* é mais antigo que *ficheiro2* (modificação)
`file1` `-ef` *`file2`*	*ficheiro1* e *ficheiro2* estão no mesmo dispositivo e no mesmo número de inode

Tabela 12.5: Lista de operadores de comparação de ficheiros na expressão condicional

Os operadores de comparação **Aritmética** de inteiros na expressão regular são ”`-eq`”, ”`-ne`”, ”`-lt`”, ”`-le`”, ”`-gt`” e ”`-ge`”.

equação	condição para retornar o verdadeiro lógico
`-z str`	o comprimento de *str* é zero
`-n str`	o comprimento de *str* não é zero
`str1 = str2`	*str1* and *str2* são iguais
`str1 != str2`	*str1* and *str2* não são iguais
`str1 < str2`	*str1* ordena antes de *str2* (dependente do locale)
`str1 > str2`	*str1* ordena após *str2* (dependente do locale)

Tabela 12.6: Lista de operadores de comparação de cadeias na expressão condicional

12.1.4 Ciclos (loops) da shell

Existem vários idiomas de ciclo para usar na shell POSIX.

- "`for x in foo1 foo2 … ; do command ; done`" faz ciclos ao atribuir itens da lista "`foo1 foo2 …`" à variável "x" e a executar o "`comando`".
- "`while condition ; do command ; done`" repete o "`comando`" enquanto a "`condição`" for verdadeira.
- "`until condition ; do command ; done`" repete o "`comando`" enquanto a "`condição`" não for verdadeira.
- "`break`" permite sair do ciclo.
- "`continue`" permite resumir a próxima interacção do ciclo.

Dica
A interacção numérica tipo linguagem C pode ser realizada a usar seq(1) como o gerador de "`foo1 foo2 …`".

Dica
Veja Secção 9.4.9.

12.1.5 Variáveis de ambiente do shell

Algumas variáveis de ambiente populares para a linha de comandos normal da shell podem não estar disponíveis no ambiente de execução do seu script.

- Para "`$USER`", use "`$(id -un)`"
- Para "`$UID`", use "`$(id -u)`"
- Para "`$HOME`", use "`$(getent passwd "$(id -u)"|cut -d ":" -f 6)`" (isto também funciona em Secção 4.5.2)

12.1.6 A sequência de processamento da linha de comandos da shell

A shell processa um script rudemente como a seguinte sequência.

- A shell lê uma linha.
- A shell agrupa uma parte de uma linha como **um testemunho** se estiver dentro de "…" ou '…'.
- A shell divide a outra parte de uma linha em **testemunhos** como o seguinte.

  - Espaços em branco: *espaço tab nova-linha*
  - Meta-caracteres: `< > | ; & ( )`
- A shell verifica a **palavra reservada** para cada testemunho para ajustar o comportamento dele se não dentro de "…" ou '…'.
  - **palavra reservada**: `if then elif else fi for in while unless do done case esac`
- A shell expande o **alias** se não estiver dentro de "…" ou '…'.
- A shell expande o **til** se não dentro de "…" ou '…'.
  - "~" → diretório home do utilizador atual
  - "~*utilizador*" → diretório inicial do *utilizador*
- A shell expande o **parâmetro** ao seu valor se não dentro de '…'.
  - **parâmetro**: "$PARAMETER" ou "${PARAMETER}"
- A shell expande a **substituição do comando** se não dentro de '…'.
  - "`$( comando )`" → o resultado do "comando"
  - "`` ` comando ` ``" → o resultado do "comando"
- A shell expande o **glob nome_de-caminho** aos nomes de ficheiros correspondentes se não dentro de "…" ou '…'.
  - * → quaisquer caracteres
  - ? → um caractere
  - […] → qualquer um dos caracteres em "…"
- A shell procura o **comando** a partir do seguinte e executa-o.
  - definição de **função**
  - comando **builtin**
  - **ficheiro executável** em "$PATH"
- A shell vai à próxima linha e repete este processo outra vez a partir do topo desta sequência.

Citações singulares (') dentro de aspas não têm efeito.

Executar "`set -x`" na shell ou invocar a shell com a opção "`-x`" faz a shell escrever todos os comandos executados. Isto é muito útil para depuração.

12.1.7 Programas utilitários para script de shell

De modo a tornar o seu programa de shell o mais portável possível entre os sistemas Debian, é uma boa ideia limitar os programas utilitários àqueles disponibilizados pelos pacotes **essenciais**.

- "`aptitude search ~E`" lista os pacotes **essenciais**.
- "`dpkg -L nome_do-pacote |grep '/man/man.*/'`" lista as manpages (manuais) para comandos oferecidos pelo pacote *nome_do_pacote*.

Dica
Apesar de `moreutils` poder não existir fora de Debian, oferece pequenos programas interessantes. O mais notável é o sponge(8) que é bastante útil quando desejar sobrescrever o ficheiro original.

Ver Secção 1.6 exemplos.

pacote	popcon	tamanho	descrição
dash	V:929, I:999	207	shell pequeno e rápido compatível com POSIX para sh
coreutils	V:912, I:1000	17994	utilitários de núcleo GNU
grep	V:782, I:1000	1297	GNU grep, egrep e fgrep
sed	V:769, I:1000	987	GNU sed
mawk	V:495, I:998	299	awk pequeno e rápido
debianutils	V:935, I:996	228	utilitários variados específicos do Debian
bsdutils	V:421, I:999	338	utilitários básicos do 4.4BSD-Lite
bsdextrautils	V:761, I:870	382	utilitários extras do 4.4BSD-Lite
moreutils	V:18, I:39	232	utilitários Unix adicionais

Tabela 12.7: Lista de pacotes que contém programas utilitários pequenos para scripts de shell

12.2 Programação em linguagens interpretadas

pacote	popcon	tamanho	documentação
dash	V:929, I:999	207	sh: shell pequena e rápida compatível com POSIX para sh
bash	V:886, I:999	7308	sh: "info bash" fornecido por bash-doc
mawk	V:495, I:998	299	AWK: awk pequeno e rápido
gawk	V:247, I:294	3289	AWK: "info gawk" proporcionado por gawk-doc
perl	V:663, I:990	840	Perl: perl(1) e páginas html fornecidas por perl-doc e perl-doc-html
libterm-readline-gnu-perl	V:2, I:27	439	Extensão Perl para a Biblioteca GNU ReadLine/History: perlsh(1)
libreply-perl	V:0.01, I:0.07	171	REPL para Perl: reply(1)
libdevel-repl-perl	V:0.02, I:0.50	237	REPL para Perl: re.pl(1)
python3	V:720, I:975	81	Python: python3(1) e páginas html fornecidas por python3-doc
tcl	V:24, I:173	20	Tcl: tcl(3) e páginas de manual detalhadas fornecidas por tcl-doc
tk	V:18, I:167	20	Tk: tk(3) e páginas de manual detalhadas fornecidas por tk-doc
ruby	V:72, I:154	32	Ruby: ruby(1), erb(1), irb(1), rdoc(1), ri(1)

Tabela 12.8: Lista de pacotes relacionados com o interpretador

Quando deseja automatizar uma tarefa em Debian, deve primeiro fazer o script com uma linguagem interpretada. A linha de orientação para a escolha da linguagem interpretada é:

- Utilize dash, se a tarefa for simples e combinar programas CLI com um programa shell.
- Utilize python3, se a tarefa não for simples e se a estiver a escrever de raiz.
- Use perl, tcl, ruby, ... se houver um código existente usando uma dessas linguagens no Debian que precisa ser retocado para fazer a tarefa.

Se o código resultante for demasiado lento, pode reescrever apenas a parte crítica para a velocidade de execução numa linguagem compilada e chamá-la a partir da linguagem interpretada.

12.2.1 Depuração de códigos de linguagem interpretada

A maioria dos intérpretes oferece funcionalidades básicas de verificação da sintaxe e de rastreio do código.

- "**dash -n** *script.sh*" - Verificação da sintaxe de um script Shell
- "**dash -x** *script.sh*" - Rastreia um script de shell
- "**python -m py_compile** *script.py*" - Verificação da sintaxe de um script Python
- "**python -mtrace --trace** *script.py*" - Rastreia um script Python
- "**perl -I ../libpath -c** *script.pl*" - Verificação da sintaxe de um script Perl
- "**perl -d:Trace** *script.pl*" - Traça um script Perl

Para testar o código para `dash`, tente Secção 9.1.4 o que acomoda o ambiente interativo semelhante ao `bash`.

Para testar código para `perl`, tente o ambiente REPL para Perl que acomoda um ambiente REPL (=READ + EVAL + PRINT + LOOP) tipo Python para o Perl.

12.2.2 Programa GUI com o script de shell

O script shell pode ser melhorado para criar um programa GUI atrativo. O truque é usar um dos chamados programas de diálogo em vez de uma interação monótona usando comandos `echo` e `read`.

pacote	popcon	tamanho	descrição
`x11-utils`	V:228, I:555	651	xmessage(1): mostra uma mensagem ou questão numa janela (X)
`whiptail`	V:307, I:997	61	mostra caixas de diálogo amigas do utilizador a partir de scripts de shell (newt)
`dialog`	V:8, I:77	520	mostra caixas de diálogo amigas do utilizador a partir de scripts de shell (ncurses)
`zenity`	V:68, I:341	193	exibir caixas de diálogo gráficas a partir de scripts de shell (GTK)
`ssft`	V:0.01, I:0.16	80	Ferramenta Frontend de Scripts de Shell (revestimento para o zenity, kdialog e dialog com o gettext)
`gettext`	V:49, I:210	20549	"`/usr/bin/gettext.sh`": traduz mensagem

Tabela 12.9: Lista de programas de diálogo

Aqui está um exemplo de um programa GUI para demonstrar como é fácil apenas com um script de shell.

Este script usa `zenity` para selecionar um ficheiro (por defeito `/etc/motd`) e mostrá-lo.

O lançador GUI para este script pode ser criado da seguinte forma Secção 9.4.10.

```
#!/bin/sh -e
# Copyright (C) 2021 Osamu Aoki <osamu@debian.org>, Public Domain
# vim:set sw=2 sts=2 et:
DATA_FILE=$(zenity --file-selection --filename="/etc/motd" --title="Select a file to check ↩
    ") || \
  ( echo "E: File selection error" >&2 ; exit 1 )
# Check size of archive
if ( file -ib "$DATA_FILE" | grep -qe '^text/' ) ; then
  zenity --info --title="Check file: $DATA_FILE" --width 640  --height 400 \
    --text="$(head -n 20 "$DATA_FILE")"
else
  zenity --info --title="Check file: $DATA_FILE" --width 640  --height 400 \
    --text="The data is MIME=$(file -ib "$DATA_FILE")"
fi
```

Este tipo de abordagem ao programa GUI com o script de shell é útil apenas para casos de escolha simples. Se for escrever qualquer programa com complexidades, por favor considere escrevê-lo numa plataforma mais capaz.

12.2.3 Ações personalizadas para o arquivador GUI

Os programas de arquivamento GUI podem ser alargados para executar algumas ações populares em ficheiros selecionados utilizando pacotes de extensão adicionais. Também podem ser feitos para executar ações personalizadas muito específicas, adicionando os seus scripts específicos.

- Para o GNOME, consulte NautilusScriptsHowto.
- Para o KDE, veja Criando menus de serviço do Dolphin.
- Para o Xfce, consulte Thunar - Acções personalizadas e https://help.ubuntu.com/community/ThunarCustomActions.
- Para LXDE, consulte Ações personalizadas.

12.2.4 A loucura dos scripts curtos de Perl

Para processar dados, `sh` precisa gerar sub-processos executando `cut`, `grep`, `sed`, etc., e é lento. Por outro lado, o `perl` tem capacidades internas para processar dados, e é rápido. Por isso muitos scripts de manutenção do sistema na Debian usam `perl`.

Vamos pensar no seguinte trecho de script AWK de uma linha e seus equivalentes em Perl.

```
awk '($2=="1957") { print $3 }' |
```

Isto 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"' |
```

```
perl -lane 'print$F[2]if$F[1]eq+1957' |
```

Este último é um enigma. Aproveitei-me das seguintes funcionalidades do Perl.

- O espaço em branco é opcional.
- Existe a conversão automática de números para cadeia.
- Truques de execução do Perl através de opções de linha de comando: perlrun(1)
- Variáveis especiais do Perl: perlvar(1)

Esta flexibilidade é o ponto forte do Perl. Ao mesmo tempo, isto permite-nos criar códigos crípticos e emaranhados. Por isso, é preciso ter cuidado.

12.3 Codificação em linguagens compiladas

Aqui, Secção 12.3.3 e Secção 12.3.4 são incluídos para indicar como um programa semelhante a um compilador pode ser escrito em linguagem C, compilando uma descrição de nível superior em linguagem C.

pacote	popcon	tamanho	descrição
gcc	V:156, I:561	36	Compilador GNU C
libc6-dev	V:258, I:580	12863	Biblioteca GNU C: Bibliotecas de desenvolvimento e ficheiros de cabeçalho
g++	V:61, I:529	13	Compilador GNU C++
libstdc++-16-dev	V:3, I:19	24398	GNU Standard C++ Library v3 (ficheiros de desenvolvimento)
cpp	V:337, I:720	18	Pré-processador GNU C
gettext	V:49, I:210	20549	Utilitários de internacionalização GNU
glade	V:0.2, I:2.6	1613	Construtor de interfaces de utilizador GTK
valac	V:0.2, I:3.0	533	Linguagem semelhante ao C# para o sistema GObject
flex	V:6, I:67	1247	Gerador de analisador léxico rápido compatível com LEX
bison	V:7, I:71	3122	YACC-compatible gerador de análise
susv2	I:0.03	16	buscar "The Single UNIX Specifications v2"
susv3	I:0.05	16	buscar "The Single UNIX Specifications v3"
susv4	I:0.02	16	baixar "As especificações únicas UNIX v4"
golang	I:21	11	Compilador de linguagem de programação Go
rustc	V:5, I:20	13195	Linguagem de programação de sistemas Rust
gfortran	V:4, I:49	15	Compilador GNU Fortran 95
fpc	I:2.2	101	Pascal livre

Tabela 12.10: Lista de pacotes relacionados com o compilador

12.3.1 C

Pode configurar um ambiente apropriado para compilar programas escritos na linguagem de programação C com o seguinte.

```
# apt-get install glibc-doc manpages-dev libc6-dev gcc build-essential
```

O pacote `libc6-dev`, isto é, a biblioteca C GNU, disponibiliza uma biblioteca standard C a qual é uma colecção de ficheiros cabeçalho e rotinas de biblioteca usadas pela linguagem de programação C.

Veja referências para C nos seguintes.

- "`info libc`" (Referência de funções da biblioteca C)
- gcc(1) e "`info gcc`"
- each_C_library_function_name(3)
- Kernighan & Ritchie, "A Linguagem de Programação C", 2ª edição (Prentice Hall)

12.3.2 Programa C simples (gcc)

Um simples exemplo, "`example.c`" pode ser compilado com uma biblioteca "`libm`" num executável "`run_example`" com o seguinte.

```
$ 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); /* prevent buffer overflow */
        y[10] = '\0'; /* fill to make sure string ends with '\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, o "`-lm`" é necessário para ligar a biblioteca "`/usr/lib/libm.so`" do pacote `libc6` para o sqrt(3). A biblioteca real está em "`/lib/`" com o nome de ficheiro "`libm.so.6`", o qual é uma ligação simbólica para "`libm-2.7.so`".

Olhe ao último parâmetro no texto resultante. Existem mais de 10 caracteres mesmo com "`%10s`" especificado.

O uso de funções de operação de memória de ponteiro sem verificações de limites como em sprintf(3) e strcpy(3), está descontinuado para prevenir exploits de sobrelotação do buffer que influenciam os efeitos de transbordo em cima. Em vez disso, use snprintf(3) e strncpy(3).

12.3.3 Flex — um Lex melhor

O Flex é um gerador rápido de análise léxica compatível com o Lex.

O tutorial do flex(1) encontra-se em "`info flex`".

Muitos exemplos simples podem ser encontrados em "`/usr/share/doc/flex/examples/`". 1

12.3.4 Bison — um Yacc melhor

Vários pacotes disponibilizam um gerador LR parser ou LALR parser compatível em frente com o Yacc em Debian.

pacote	popcon	tamanho	descrição
`bison`	V:7, I:71	3122	gerador de análise GNU LALR
`byacc`	V:0.1, I:2.9	263	Gerador de análise Berkeley LALR
`btyacc`	V:0.01, I:0.04	247	gerador de análises de retrocesso baseado no `byacc`

Tabela 12.11: Lista de geradores de análise LALR compatíveis com Yacc

O tutorial para o bison(1) encontra-se em "`info bison`".

Tem de disponibilizar as suas próprias chamadas "`main()`" e "`yyerror()`". "`main()`" chama "`yyparse()`" que chama "`yylex()`", geralmente criada com Flex.

Aqui está um exemplo para criar um programa simples de calculadora de terminal.

Vamos criar `example.y`:

```
/* calculator source for bison */
%{
#include <stdio.h>
extern int yylex(void);
extern int yyerror(char *);
%}
```

1Poderão ser necessárias algumas adaptações para que funcionem com o sistema atual.

```
/* declare tokens */
%token NUMBER
%token OP_ADD OP_SUB OP_MUL OP_RGT OP_LFT OP_EQU

%%
calc:
 | calc exp OP_EQU    { printf("Y: RESULT = %d\n", $2); }
 ;

exp: factor
 | exp OP_ADD factor  { $$ = $1 + $3; }
 | exp OP_SUB factor  { $$ = $1 - $3; }
 ;

factor: term
 | factor OP_MUL term { $$ = $1 * $3; }
 ;

term: NUMBER
 | OP_LFT exp OP_RGT  { $$ = $2; }
  ;
%%

int main(int argc, char **argv)
{
  yyparse();
}

int yyerror(char *s)
{
  fprintf(stderr, "error: '%s'\n", s);
}
```

Vamos criar, `example.l`:

```
/* calculator source for flex */
%{
#include "example.tab.h"
%}

%%
[0-9]+ { printf("L: NUMBER = %s\n", yytext); yylval = atoi(yytext); return NUMBER; }
"+"    { printf("L: OP_ADD\n"); return OP_ADD; }
"-"    { printf("L: OP_SUB\n"); return OP_SUB; }
"*"    { printf("L: OP_MUL\n"); return OP_MUL; }
"("    { printf("L: OP_LFT\n"); return OP_LFT; }
")"    { printf("L: OP_RGT\n"); return OP_RGT; }
"="    { printf("L: OP_EQU\n"); return OP_EQU; }
"exit" { printf("L: exit\n");   return YYEOF; } /* YYEOF = 0 */
.      { /* ignore all other */ }
%%
```

Em seguida, execute o seguinte a partir do prompt do shell para tentar isso:

```
$ bison -d example.y
$ flex example.l
$ gcc -lfl example.tab.c lex.yy.c -o example
$ ./example
1 + 2 * ( 3 + 1 ) =
L: NUMBER = 1
L: OP_ADD
```

```
L: NUMBER = 2
L: OP_MUL
L: OP_LFT
L: NUMBER = 3
L: OP_ADD
L: NUMBER = 1
L: OP_RGT
L: OP_EQU
Y: RESULT = 9

exit
L: exit
```

12.4 Ferramentas de análise de código estático

As ferramentas do tipo Lint podem ajudar automaticamente a análise de código estático.

As ferramentas do tipo indentação podem ajudar as revisões de código humano, reformatando os códigos-fonte de forma consistente.

As ferramentas do tipo Ctags podem ajudar nas revisões humanas de código, gerando um ficheiro de índice (ou tag) de nomes encontrados nos códigos fonte.

Dica
Configurar o seu editor favorito (`emacs` ou `vim`) para usar plugins assíncronos do motor lint, ajuda-o a escrever o seu código. Estes plugins estão a ficar muito poderosos tirando partido do Protocolo do Servidor de Idiomas. Como eles estão evoluindo rapidamente, usar o seu código original ao invés do pacote Debian pode ser uma boa opção.

12.5 Depuração

A depuração é uma parte importante das atividades de programação. Saber como depurar programas faz de si um bom utilizador de Debian capaz de produzir relatórios de bugs significantes.

12.5.1 Execução gdb básica

O depurador principal em Debian é o gdb(1) que lhe permite inspeccionar um programa enquanto ele é executado.

Vamos instalar o `gdb` e programas relacionados com o seguinte.

```
# apt-get install gdb gdb-doc build-essential devscripts
```

Pode ser encontrado um bom tutorial do `gdb`:

- “`info gdb`”
- ”Depurando com GDB” em `/usr/share/doc/gdb-doc/html/gdb/index.html`
- ”tutorial na web”

Aqui está um exemplo simples de utilização do gdb(1) num ”`program`” compilado com a opção ”`-g`” para produzir informação de depuração.

pacote	popcon	tamanho	descrição
vim-ale	I:0.74	2833	Mecanismo assíncrono Lint para Vim 8 e NeoVim
vim-syntastic	I:1.9	1379	Truques de verificação de sintaxe para o vim
elpa-flycheck	V:0.1, I:1.6	1130	verificação de sintaxe moderna em tempo real para o Emacs
elpa-relint	V:0.00, I:0.04	150	Localizador de erros de regexp do Emacs Lisp
cppcheck-gui	V:0.11, I:0.97	7973	ferramenta para análise estática de código C/C++ (GUI)
shellcheck	V:4, I:17	22860	ferramenta lint para scripts shell
pyflakes3	V:2, I:14	20	verificador passivo de programas Python 3
pylint	V:4, I:19	2104	Verificador estático de código Python
perl	V:663, I:990	840	interpretador com verificador de código estático interno: `B::Lint`(3perl)
rubocop	V:0.1, I:1.0	4502	Analisador de código estático Ruby
clang-tidy	V:2, I:11	20	ferramenta de verificação de formato de regras C++ baseada em Clang
splint	V:0.05, I:0.79	2325	ferramenta para verificação estática de programas C por bugs
flawfinder	V:0.05, I:0.43	187	ferramenta para examinar código fonte C/C++ e procurar por fraquezas na segurança
black	V:3, I:15	5868	formatador de código Python sem compromissos
perltidy	V:0.4, I:2.9	3086	Indentador e reformatador de scripts Perl
indent	V:0.3, I:4.7	438	Programa de formatação de código fonte em linguagem C
astyle	V:0.2, I:2.5	770	Indentador de código-fonte para C, C++, Objective-C, C# e Java
bcpp	V:0.02, I:0.22	114	Embelezador C(++)
xmlindent	V:0.07, I:0.73	52	Reformatador de fluxo XML
global	V:0.2, I:1.5	1923	Ferramentas de pesquisa e navegação de código-fonte
exuberant-ctags	V:1, I:12	341	criar índices de ficheiros de etiquetas de definições de código-fonte
universal-ctags	V:1, I:12	4238	criar índices de ficheiros de etiquetas de definições de código-fonte

Tabela 12.12: Lista de ferramentas para análise de código estático

pacote	popcon	tamanho	documentação
gdb	V:92, I:162	13472	”`info gdb`” disponibilizado por `gdb-doc`
ddd	V:0.3, I:4.9	5094	”`info ddd`” disponibilizado por `ddd-doc`

Tabela 12.13: Lista de pacotes de depuração

```
$ gdb program
(gdb) b 1               # set break point at line 1
(gdb) run args          # run program with args
(gdb) next              # next line
...
(gdb) step              # step forward
...
(gdb) p parm            # print parm
...
(gdb) p parm=12         # set value to 12
...
(gdb) quit
```

Dica
Muitos comandos do gdb(1) podem ser abreviados. A expansão da Tab funciona como na shell.

12.5.2 Depurar o pacote Debian

Uma vez que todos os binários instalados devem ser despojados no sistema Debian por defeito, a maioria dos símbolos de depuração são removidos no pacote normal. De modo a depurar pacotes Debian com gdb(1), os pacotes `*-dbgsym` precisam de ser instalados (e.g. `coreutils-dbgsym` no caso do `coreutils`). Os pacotes fonte geram pacotes `*-dbgsym` automaticamente juntamente com os pacotes binários normais e esses pacotes de depuração são colocados separadamente no arquivo debian-debug. Por favor, consulte os artigos no Debian Wiki para mais informações.

Se um pacote a ser depurado não fornecer o seu pacote `*-dbgsym`, é necessário instalá-lo depois de o reconstruir através do seguinte.

```
$ mkdir /path/new ; cd /path/new
$ sudo apt-get update
$ sudo apt-get dist-upgrade
$ sudo apt-get install fakeroot devscripts build-essential
$ apt-get source package_name
$ cd package_name*
$ sudo apt-get build-dep ./
```

Corrigir bugs se necessário.

Mude a versão de pacote para uma que não colida com as versões oficiais de Debian, por exemplo, uma adicionada com "`+debug1`" quando se recompila uma versão de pacote existente, ou uma adicionada com "`~pre1`" quando se compila uma versão de pacote ainda não lançada com o seguinte.

```
$ dch -i
```

Compilar e instalar pacotes com símbolos de depuração com o seguinte.

```
$ export DEB_BUILD_OPTIONS="nostrip noopt"
$ debuild
$ cd ..
$ sudo debi package_name*.changes
```

Necessita verificar os scripts de construção do pacote e assegurar o uso de "`CFLAGS=-g -Wall`" para compilar binários.

12.5.3 Obter um backtrace

Quando encontrar uma falha num programa, é uma boa ideia enviar um relatório de bug com informação de rastreio copiada-e-colada.

O backtrace pode ser obtido pelo gdb(1) utilizando uma das seguintes abordagens:

- Abordagem às falhas em GDB:
  - Execute o programa a partir do GDB.
  - Falha no programa.
  - Digite "`bt`" no prompt da GDB.
- Abordagem falhas primeiro:
  - Actualize o ficheiro "**/etc/security/limits.conf**" para incluir o seguinte:

```
* soft core unlimited
```

  - Escreva "`ulimit -c unlimited`" na prompt da shell.
  - Execute o programa a partir desta janela de comandos.
  - Colapsa o programa para produzir um ficheiro de despejo do núcleo.
  - Carregar o ficheiro core dump para GDB como "`gdb gdb ./program_binary core`" .
  - Digite "`bt`" no prompt da GDB.

Para uma situação de loop infinito ou de teclado congelado, pode forçar o encerramento do programa premindo `Ctrl-\` ou `Ctrl-C` ou executando "`kill -ABRT` *PID*". (Ver Secção 9.4.12)

Dica
Muitas vezes, vê um backtrace onde uma ou mais linhas do topo estão em "`malloc()`" ou "`g_malloc()`". Quando isto acontece, há grandes hipóteses do seu backtrace não ser muito útil. O modo mais fácil de encontrar alguma informação útil é definir a variável de ambiente "`$MALLOC_CHECK_`" para um valor de 2 (malloc(3)). Pode fazer isto enquanto corre o `gdb` ao fazer o seguinte.

```
$ MALLOC_CHECK_=2 gdb hello
```

12.5.4 Comandos gdb avançados

comando	descrição dos objetivos do comando
`(gdb) thread apply all bt`	obter um backtrace para todos os processos de um programa de multi-processo
`(gdb) bt full`	obter parâmetros que vêm na pilha das chamadas de função
`(gdb) thread apply all bt full`	obtém um backtrace e parâmetros como a combinação das opções precedentes
`(gdb) thread apply all bt full 10`	obter um backtrace e parâmetros para as 10 chamadas do topo para cortar resultados irrelevantes
`(gdb) set logging on`	escreve um relatório dos resultados do `gdb` para um ficheiro (a predefinição é "`gdb.txt`")

Tabela 12.14: Lista de comandos gdb avançados

12.5.5 Verificar a dependência em bibliotecas

Use o ldd(1) para encontrar uma dependência de um programa em bibliotecas com o seguinte.

```
$ ldd /usr/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 que o ls(1) funcione num ambiente 'chroot', as bibliotecas em cima têm de estar disponíveis no seu ambiente 'chroot'.

Veja Secção 9.4.6.

12.5.6 Ferramentas dinâmicas de rastreio de chamadas

Existem várias ferramentas de rastreio de chamadas dinâmicas disponíveis em Debian. Veja Secção 9.4.

12.5.7 Depurar Erros do X

Se o programa do GNOME `preview1` recebeu um erro do X, deverá ver a mensagem que a seguir.

```
The program 'preview1' received an X Window System error.
```

Neste caso, pode tentar correr o programa com "`--sync`" e quebrar a função "`gdk_x_error`" de modo a obter um backtrace.

12.5.8 Ferramentas de detecção de fugas de memória

Aqui estão várias ferramentas de detecção de fugas de memória em Debian.

pacote	popcon	tamanho	descrição
`libc6-dev`	V:258, I:580	12863	mtrace(1): funcionalidades de depuração do malloc em glibc
`valgrind`	V:4, I:32	89318	depurador e perfilador de memória
`electric-fence`	V:0.2, I:2.2	69	o depurador malloc(3)
`libdmalloc5`	V:0.01, I:0.57	373	biblioteca de depuração de alocação de memória
`duma`	V:0.00, I:0.05	297	biblioteca para detetar "overruns" e "under-runs" de buffer em programas C e C++
`leaktracer`	V:0.01, I:0.27	56	rastreador de fugas de memória para programas C++

Tabela 12.15: Lista de ferramentas de detecção de fugas de memória

12.5.9 Desassemblar binário

Pode desassemblar código binário com o objdump(1) com o seguinte.

```
$  objdump -m i386 -b binary -D /usr/lib/grub/x86_64-pc/stage1
```

Nota
O gdb(1) pode ser usado para desmontar (desassemblar) código interativamente.

12.6 Ferramentas de construção

pacote	popcon	tamanho	documentação
make	V:145, I:562	1762	"info make" disponibilizado por make-doc
autoconf	V:27, I:193	2244	"info autoconf" disponibilizado por autoconf-doc
automake	V:26, I:192	1958	"info automake" disponibilizado por automake1.10-doc
libtool	V:23, I:176	1276	"info libtool" fornecido por libtool-doc
cmake	V:22, I:123	59735	cmake(1) sistema make multiplataforma e de código aberto
ninja-build	V:8, I:54	456	ninja(1) pequeno sistema de construção mais próximo em espírito do Make
meson	V:6, I:29	4402	meson(1) sistema de construção de alta produtividade no topo do ninja
xutils-dev	V:0.6, I:6.7	1495	imake(1), xmkmf(1), etc.

Tabela 12.16: Lista de pacotes de ferramentas de compilação

12.6.1 Make

O Make é um utilitário para manutenção de grupos de programas Após a execução do make(1), o make lê o ficheiro de regras, "Makefile" e atualiza um alvo se depender de ficheiros pré-requisitados que foram modificados desde que o alvo foi modificado por último, ou se o alvo não existir. A execução destas atualizações pode ocorrer concorrentemente.

A regra de sintaxe do ficheiro é a seguinte.

```
target: [ prerequisites ... ]
 [TAB]  command1
 [TAB]  -command2 # ignore errors
 [TAB]  @command3 # suppress echoing
```

Aqui "[TAB]" é um código TAB. Cada linha é interpretada pela shell após fazer a substituição da variável. Utilize "\" no final de uma linha para continuar o script. Use "$$" para inserir "$" para valores de ambiente para um script de shell.

Podem ser escritas regras implícitas para o destino e pré-requisitos, por exemplo, com o seguinte.

```
%.o: %.c header.h
```

Aqui, o alvo contém o caractere "%" (exatamente um deles). O "%" pode corresponder a qualquer subcadeia não vazia nos nomes de ficheiros do próprio alvo. Os pré-requisitos usam igualmente "%" para mostrar como os seus nomes estão relacionados ao próprio nome do alvo.

variável automática	valor
$@	alvo
$<	primeiro pré-requisito
$?	todos os novos pré-requisitos
$^	todos os pré-requisitos
$*	"%" estaminal correspondente no modelo de destino

Tabela 12.17: Lista de variáveis automáticas do make

Corra "make -p -f/dev/null" para ver as regras internas automáticas.

expansão da variável	descrição
`foo1 := bar`	expansão de uma vez
`foo2 = bar`	expansão recursiva
`foo3 += bar`	acrescentar

Tabela 12.18: Lista de expansões da variável do make

12.6.2 Autotools

O Autotools é um conjunto de ferramentas de programação concebido para ajudar a tornar os pacotes de código-fonte portáveis para muitos sistemas do tipo Unix.

- O Autoconf é uma ferramenta para produzir um script shell "`configure`" a partir de "`configure.ac`".
  - O "`configure`" é utilizado mais tarde para produzir o"`Makefile`" a partir do modelo "`Makefile.in`".
- Automake é uma ferramenta para produzir "`Makefile.in`" a partir de "`Makefile.am`".
- Libtool é um script shell para resolver o problema de portabilidade de software ao compilar bibliotecas partilhadas a partir do código fonte.

12.6.2.1 Compilar e instalar um programa



Atenção
Não substitua ficheiros do sistema com os seus programas compilados quando os instalar.

Debian não toca nos ficheiros em "`/usr/local/`" ou em "`/opt`". Portanto se compilar um programa a partir do código-fonte, instale-o em "`/usr/local/`" para que não interfira com o Debian.

```
$ cd src
$ ./configure --prefix=/usr/local
$ make # this compiles program
$ sudo make install # this installs the files in the system
```

12.6.2.2 Desinstalar um programa

Se tiver o código original, se ele utiliza autoconf(1)/automake(1) e se você lembrar como o configurou, execute como segue para desinstalar o programa.

```
$ ./configure all-of-the-options-you-gave-it
$ sudo make uninstall
```

Em alternativa, se tiver a absoluta certeza que o processo de instalação apenas coloca ficheiros sob "`/usr/local/`" e não há lá nada importante, pode apagar todos os seus conteúdos com o seguinte.

```
# find /usr/local -type f -print0 | xargs -0 rm -f
```

Se não tiver a certeza de onde os ficheiros estão instalados, deve considerar usar o checkinstall(8) do pacote `checkinstall`, que disponibiliza um caminho limpo para a desinstalação. Agora suporta criar um pacote Debian com a opção "`-D`".

12.6.3 Meson

O sistema de construção de software tem vindo a evoluir:

- Autotools no topo do Make tem sido o padrão de facto para a infraestrutura de compilação portátil desde os anos 1990. Isso é extremamente lento.
- O CMake, inicialmente lançado em 2000, melhorou significativamente a velocidade, mas foi originalmente construído sobre o inerentemente lento Make. (Agora Ninja pode ser seu backend.)
- O Ninja, lançado inicialmente em 2012, destina-se a substituir o Make para melhorar a velocidade de construção e foi concebido para que os seus ficheiros de entrada sejam gerados por um sistema de construção de nível superior.
- O Meson, lançado inicialmente em 2013, é o novo sistema de construção de nível superior popular e rápido que usa o Ninja como backend.

Ver documentos encontrados em "O sistema de construção Meson" e "O sistema de construção Ninja".

12.7 Web

Páginas web dinâmicas interactivas podem ser feitas conforme a seguir.

- As questões são apresentadas ao explorador do utilizador a usar formulários HTML.
- Preencher e clicar nas entradas do formulário envia uma das seguintes cadeias de URL com parâmetros codificados do explorador para o servidor web.
  - "`https://www.foo.dom/cgi-bin/program.pl?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3`"
  - "`https://www.foo.dom/cgi-bin/program.py?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3`"
  - "`https://www.foo.dom/program.php?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3`"
- O "%nn" no URL é substituído por um caractere com valor hexadecimal `nn`.
- A variável de ambiente está definida como: "`QUERY_STRING="VAR1=VAL1 VAR2=VAL2 VAR3=VAL3"`".
- O programa CGI (qualquer um de "`program.*`") no servidor web executa-se a si próprio com a variável de ambiente "`$QUERY_STRING`".
- O `stdout` do programa CGI é enviado ao explorador web e é apresentado como uma página web dinâmica e interactiva.

Por razões de segurança é melhor não embarcar em novos hacks para analisar parâmetros CGI. Existem módulos definidos para eles em Perl e Python. O PHP vem com estas funcionalidades. Quando é necessário o armazenamento de dados no cliente, usam-se cookies HTTP. Quando é necessário o processamento de dados no lado do cliente, usa-se frequentemente Javascript.

Para mais, veja Common Gateway Interface, The Apache Software Foundation e JavaScript.

Procurar "CGI tutorial" no Google ao escrever diretamente o URL codificado https://www.google.com/search?hl=en&ie=UTF-8&q=CGI+tutorial no endereço do explorador é uma boa maneira de ver o script CGI em acção no servidor da Google.

12.8 A tradução do código-fonte

Existem programas para converter código-fonte.

pacote	popcon	tamanho	palavra chave	descrição
perl	V:663, I:990	840	AWK → PERL	converter código-fonte de AWK para PERL: a2p(1)
f2c	V:0.1, I:1.9	443	FORTRAN → C	converter código-fonte de FORTRAN 77 para C/C++: f2c(1)
intel2gas	V:0.00, I:0.16	178	NASM → GAS	conversor de NASM (formato Intel) ao GNU Assembler (GAS)

Tabela 12.19: Lista de ferramentas de tradução de código-fonte

12.9 Criar um pacote Debian

Se desejar criar um pacote Debian, leia o seguinte.

- Capítulo 2 para compreender o sistema básico de pacotes
- Secção 2.7.13 para compreender o processo básico de portar
- Secção 9.11.4 para compreender as técnicas de chroot básicas
- debuild(1), e sbuild(1)
- Secção 12.5.2 para recompilar para depuração
- Guia dos Novos Maintainers da Debian (o pacote `debmake-doc`)
- Referência de Programadores da Debian (o pacote `developers-reference`)
- Manual de Políticas Debian (o pacote `debian-policy`)

Existem pacotes como os `debmake`, `dh-make`, `dh-make-perl`, etc., que auxiliam no processo em empacotamento.

Apêndice A

Apêndice

Aqui estão as origens deste documento.

A.1 o labirinto Debian

O sistema Linux é uma plataforma de computação muito poderosa para um computador em rede. No entanto, aprender a usar todas as suas capacidades não é fácil. Configurar a lista de trabalhos de impressora LPR com uma impressora não-PostScript era um bom exemplo para tropeçar. (Não existe mais esse problema porque as instalações recentes usam o novo sistema CUPS.)

Existe um mapa completo e detalhado chamado o ”CÓDIGO FONTE”. Este é muito preciso mas muito difícil de compreender. Existem também referências chamadas HOWTO e mini-HOWTO. São mais fáceis de compreender mas tendem a dar-lhe demasiados detalhes e a perder o objetivo principal. Por vezes tenho problemas a encontrar a secção correcta num HOWTO longo quando preciso de invocar alguns comandos.

Espero que este ”Debian Reference (versão 2.152)” (2026-10-10 14:56:25 UTC) disponibilize um bom ponto de partida para pessoas no labirinto Debian.

A.2 História do Copyright

O Debian Reference foi iniciado por mim, Osamu Aoki <osamu at debian dot org> como um memo pessoal de administração do sistema. Muitos conteúdos vieram do conhecimento que ganhei a partir da lista de email debian-user e de outros recursos Debian.

A seguir uma sugestão de Josip Rodin, que estava muito ativo com o Projecto de Documentação de Debian (DDP), o ”Debian Reference (versão 1, 2001-2007)” foi criado como parte dos documentos DDP.

Após 6 anos, percebi que a ”Debian Reference (versão 1)” estava ultrapassada e comecei a reescrever muitos dos seus conteúdos. A nova ”Debian Reference (versão 2)” é lançada em 2008.

Eu atualizei a ”Referência Debian (versão 2)” para abordar novos tópicos (Systemd, Wayland, IMAP, PipeWire, kernel Linux 5.10) e removi tópicos desatualizados (SysV init, CVS, Subversion, protocolo SSH 1, kernels Linux antes do 2.5). Referências a Jessie 8 (2015-2020) situações de lançamento ou mais antigas são na sua maioria removidas.

Esta ”Referência Debian (versão 2.152)” (2026-10-10 14:56:25 UTC) cobre maioritariamente os lançamentos Debian Trixie (`=stable`) e Forky (`=testing`).

Os conteúdos do tutorial pode ser rastreados até à sua origem e inspiração com o seguinte.

- ”Guia do Utilizador de Linux” por Larry Greenfield (Dezembro 1996)

  - tornado obsoleto pelo ”Debian Tutorial”
- ”Tutorial Debian” por Havoc Pennington. (11 de Dezembro, 1998)
  - parcialmente escrito por Oliver Elphick, Ole Tetlie, James Treacy, Craig Sawyer e Ivan E. Moore II
  - tornado obsoleto por ”Debian GNU/Linux: Guia de Instalação e Utilização”
- ”Debian GNU/Linux: Guia de Instalação e Utilização” por John Goerzen e Ossama Othman (1999)
  - tornado obsoleto pela ”Debian Reference (versão 1)”

A descrição do pacote e do arquivo podem rastrear alguma da origem e inspiração dele no seguinte.

- ”FAQ do Debian” (versão de Março 2002, quando era mantido por Josip Rodin)

Os outros conteúdos podem rastrear alguma da origem e inspiração deles no seguinte.

- ”Referência Debian (versão 1)” por Osamu Aoki (2001–2007)
  - tornado obsoleto pela nova ”Debian Reference (versão 2)” em 2008.

A ”Debian Reference (versão 1)” anterior foi criada com muitos contribuidores.

- a maior contribuição de conteúdo de tópicos de configuração de rede por Thomas Hood
- contribuição significativa de conteúdos em tópicos relacionados com X e VCS por Brian Nelson
- a ajuda na construção de scripts de compilação e muitas correcções no conteúdo por Jens Seidel
- revisão extensiva por David Sewell
- muitas contribuições pelos tradutores, contribuidores e relatórios de bugs

Muitas páginas de manual e páginas de informação sobre o sistema Debian, assim como páginas web a montante e documentos da Wikipedia foram usados como referências primárias para escrever este documento. Na medida em que Osamu Aoki considerou dentro do uso justo, muitas partes delas, especialmente definições de comandos, foram usadas como peças de frase após cuidadosos esforços editoriais para encaixá-las no estilo e no objetivo deste documento.

A descrição do depurador gdb foi expandida a usar Conteúdos wiki Debian de backtrace com consentimento de Ari Pollak, Loïc Minier e Dafydd Harries.

O conteúdo do ”Debian Reference (versão 2.152)” (2026-10-10 14:56:25 UTC) é maioritariamente trabalho meu com excepção do descrito acima. Este foi também atualizado pelos contribuidores.

O autor, Osamu Aoki, agradece a todos os que ajudaram a tornar possível este documento.

A.3 Formato do documento

A fonte do documento original em inglês está atualmente escrita em ficheiros DocBook XML. Esta fonte Docbook XML é convertida em HTML, texto simples, PostScript e PDF. (Alguns formatos podem ser ignorados para distribuição.)