



debian

Referência Debian

Osamu Aoki

Copyright © 2013-2021 Osamu Aoki

Esta Referência Debian (versão 2.87) (2021-11-03 08:37:36 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.

COLLABORATORS

	<i>TITLE :</i> Referência Debian		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Osamu Aoki	3 de novembro de 2021	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Conteúdo

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	The shell prompt under 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	3
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	6
1.2.2	Internos do sistema de ficheiros	7
1.2.3	Permissões do sistema de ficheiros	8
1.2.4	Controlo de permissões para ficheiros acabados de criar: umask	10
1.2.5	Permissões para grupos de utilizadores (group)	11
1.2.6	Marcas temporais (Timestamps)	12
1.2.7	Links (ligações)	13
1.2.8	Pipes com nome (FIFOs)	14
1.2.9	Sockets	14
1.2.10	Ficheiros de aparelho	15
1.2.11	Ficheiros de aparelhos especiais	15
1.2.12	procfs e sysfs	16
1.2.13	tmpfs	16
1.3	Midnight Commander (MC)	17

1.3.1	Personalização do MC	17
1.3.2	Iniciar o MC	17
1.3.3	Gestor de ficheiros no MC	17
1.3.4	Truques de linha de comandos no MC	18
1.3.5	O editor interno em MC	18
1.3.6	O visualizador interno no MC	19
1.3.7	Funcionalidades de auto-arranque do MC	19
1.3.8	Sistema de ficheiros virtual FTP no MC	19
1.4	O ambiente de trabalho estilo Unix básico	19
1.4.1	A shell de login	19
1.4.2	Personalizar bash	20
1.4.3	Teclas especiais	21
1.4.4	Mouse operations	21
1.4.5	O pager	22
1.4.6	O editor de texto	22
1.4.7	Definir um editor de texto predefinido	22
1.4.8	Using vim	23
1.4.9	Gravar as atividades da shell	23
1.4.10	Comandos básicos de Unix	24
1.5	O comando simples da shell	26
1.5.1	Execução do comando e variável de ambiente	26
1.5.2	A variável "\$LANG"	26
1.5.3	A variável "\$PATH"	27
1.5.4	A variável "\$HOME"	28
1.5.5	Opções da linha de comandos	28
1.5.6	Glob da shell	28
1.5.7	Valor de retorno do comando	29
1.5.8	Sequências de comandos típicas e redireccionamento da shell	29
1.5.9	Comando alias	31
1.6	Processamento de texto estilo Unix	32
1.6.1	Ferramentas de texto de Unix	32
1.6.2	Expressões regulares	33
1.6.3	Expressões de substituição	33
1.6.4	Substituição global com expressões regulares	35
1.6.5	Extrair dados de tabela de ficheiro de texto	36
1.6.6	Trechos de script para canalizar comandos em pipe	37

2	Gestão de pacotes Debian	38
2.1	Pré-requisitos da gestão de pacotes Debian	38
2.1.1	Configuração de pacotes	38
2.1.2	Precauções básicas	39
2.1.3	A vida com atualizações eternas	40
2.1.4	Básico do arquivos Debian	41
2.1.5	Debian é 100% software livre	44
2.1.6	Dependências de pacote	45
2.1.7	O fluxo de eventos da gestão de pacotes	46
2.1.8	Primeira resposta a problemas com a gestão de pacotes	47
2.2	Operações básicas de gestão de pacotes	47
2.2.1	<code>apt</code> vs. <code>apt-get</code> / <code>apt-cache</code> contra o <code>aptitude</code>	47
2.2.2	Operações básicas de gestão de pacotes com a linha de comandos	50
2.2.3	Uso interativo do <code>aptitude</code>	50
2.2.4	Teclas de atalho do <code>aptitude</code>	51
2.2.5	Vistas de pacote no <code>aptitude</code>	51
2.2.6	Opções do método de pesquisa com o <code>aptitude</code>	52
2.2.7	A fórmula regex do <code>aptitude</code>	53
2.2.8	Resolução de dependências do <code>aptitude</code>	53
2.2.9	Relatórios (logs) de atividade de pacotes	55
2.3	Exemplos de operações do <code>aptitude</code>	55
2.3.1	Listagem de pacotes com correspondência por expressão regular nos nomes de pacotes	55
2.3.2	Explorar com a correspondência de expressão regular	55
2.3.3	Purgar pacotes removidos definitivamente	55
2.3.4	Acertar o estado auto/manual de instalação	56
2.3.5	atualização total ao sistema	56
2.4	Operações de gestão avançada de pacotes	57
2.4.1	Operações de gestão avançada de pacotes com linha de comandos	57
2.4.2	Verificação dos ficheiros pacotes instalados	59
2.4.3	Salvaguardar para problemas de pacotes	59
2.4.4	Procurar nos meta-dados do pacote	60
2.5	Os interiores da gestão de pacotes Debian	60
2.5.1	Meta dados do arquivo	60
2.5.2	Ficheiro "Release" de nível de topo e autenticidade:	60
2.5.3	Ficheiros "Release" do nível de arquivo	61
2.5.4	Obter os meta dados do pacote	62
2.5.5	O estado dos pacote para o APT	63
2.5.6	O estado de pacotes para o <code>aptitude</code>	63
2.5.7	Cópias locais dos pacotes obtidos	63

2.5.8	Nomes de ficheiros de pacotes Debian	63
2.5.9	O comando <code>dpkg</code>	64
2.5.10	O comando <code>update-alternatives</code>	65
2.5.11	O comando <code>dpkg-statoverride</code>	65
2.5.12	O comando <code>dpkg-divert</code>	65
2.6	Recuperação de um sistema danificado	65
2.6.1	Incompatibilidade com configurações antigas de utilizador	66
2.6.2	Pacotes diferentes com ficheiros sobrepostos	66
2.6.3	Corrigir script problemático de pacote	66
2.6.4	Recuperação com o comando <code>dpkg</code>	67
2.6.5	Recuperar dados de seleção de pacotes	67
2.7	Dicas para a gestão de pacotes	68
2.7.1	Como escolher os pacotes Debian	68
2.7.2	Pacotes de fontes de arquivos misturados	68
2.7.3	Moldar a versão candidata	70
2.7.4	atualizações e Backports	71
2.7.5	Bloquear pacotes instalados por "Recomendados"	72
2.7.6	Acompanhar <code>testing</code> com alguns pacotes de <code>unstable</code>	72
2.7.7	Acompanhar <code>unstable</code> com alguns pacotes de <code>experimental</code>	73
2.7.8	Descarga e atualização automática de pacotes	74
2.7.9	Limitar a largura de banda de descarga para o APT	74
2.7.10	Downgrade de emergência	74
2.7.11	Quem fez o upload do pacote?	75
2.7.12	O pacote <code>equivs</code>	75
2.7.13	Portar um pacote ao sistema <code>stable</code>	76
2.7.14	Servidor proxy para o APT	76
2.7.15	Mais leituras sobre a gestão de pacotes	77
3	A inicialização do sistema	78
3.1	Uma visão geral do processo de arranque	78
3.1.1	Stage 1: the UEFI	78
3.1.2	Estágio 2: o gestor de arranque	79
3.1.3	Estágio 3: o mini-sistema Debian	80
3.1.4	Estágio 4: o sistema Debian normal	81
3.2	<code>init</code> do <code>Systemd</code>	81
3.2.1	O nome da máquina	83
3.2.2	O sistema de ficheiros	83
3.2.3	Inicialização da interface de rede	83
3.3	A mensagem do kernel	83

3.4	A mensagem do sistema	83
3.5	System management	84
3.6	Other system monitors	84
3.7	Personalizar o systemd	84
3.7.1	Socket activation	84
3.8	O sistema udev	86
3.8.1	A inicialização de módulos do kernel	87
4	Authentication and access controls	88
4.1	Autenticação normal de Unix	88
4.2	Gerir informação de conta e palavra-passe	90
4.3	Boa palavra-passe	90
4.4	Criar palavra-passe encriptada	91
4.5	PAM e NSS	91
4.5.1	Ficheiros de configuração acedidos pelo PAM e NSS	92
4.5.2	O moderno sistema de gestão centralizado	92
4.5.3	”Porque o su do GNU não suporta o grupo wheel”	93
4.5.4	Regras de palavra-passe rigorosas	93
4.6	Segurança da autenticação	94
4.6.1	Palavra-passe segura na Internet	94
4.6.2	Shell Segura	94
4.6.3	Medidas de segurança extra para a Internet	94
4.6.4	Tornar a palavra-passe do root segura	95
4.7	Outros controles de acesso	95
4.7.1	sudo	96
4.7.2	PolicyKit	96
4.7.3	Recadeiar acesso a alguns serviços de servidor	96
4.7.4	Linux security features	97
5	Configuração de rede	98
5.1	A infra-estrutura de rede básica	98
5.1.1	A resolução de nome de máquina	98
5.1.2	O nome da interface de rede	100
5.1.3	A gama de endereços de rede para a LAN	101
5.1.4	O suporte a aparelhos de rede	101
5.2	A configuração moderna de rede para desktop	101
5.2.1	Ferramentas GUI de configuração de rede	102
5.3	A moderna configuração de rede sem GUI	102
5.4	A configuração de rede de baixo nível	103

5.4.1	Comandos iproute2	103
5.4.2	Operações de rede seguras de baixo nível	103
5.5	Optimização da rede	104
5.5.1	Encontrar o MTU óptimo	104
5.5.2	Optimização WAN TCP	105
5.6	Infraestrutura netfilter	105
6	Aplicações de rede	107
6.1	Navegadores web	107
6.1.1	Configuração do explorador	108
6.2	O sistema de correio electrónico (mail)	108
6.2.1	Noções básicas de mail	109
6.2.2	Modern mail service limitation	109
6.2.3	Historic mail service expectation	110
6.2.4	Agente de transporte de mail (MTA)	110
6.2.4.1	A configuração do exim4	110
6.2.4.2	A configuração do postfix com SASL	112
6.2.4.3	A configuração do endereço de mail	113
6.2.4.4	Operações MTA básicas	114
6.3	O servidor de acesso remoto e utilitários (SSH)	115
6.3.1	Bases do SSH	115
6.3.2	User name on the remote host	116
6.3.3	Ligar sem palavras-passe remotas	116
6.3.4	Lidar com clientes SSH alienígenas	116
6.3.5	Configurar o ssh-agent	117
6.3.6	Sending a mail from a remote host	117
6.3.7	Reencaminhamento de portos para SMTP/POP3 em túnel	117
6.3.8	Como desligar o sistema remoto em SSH	117
6.3.9	Depurar problemas no SSH	118
6.4	O servidor de impressão e utilitários	118
6.5	Outras aplicações de servidor de rede	118
6.6	Outros clientes de aplicação de rede	119
6.7	Os diagnósticos dos daemons do sistema	119
7	GUI System	121
7.1	GUI desktop environment	121
7.2	GUI communication protocol	122
7.3	GUI infrastructure	123
7.4	GUI applications	123

7.5	Fonts	123
7.5.1	Fontes (tipos de letra) básicas	125
7.5.2	Font rasterization	126
7.6	Sandbox	127
7.7	Remote desktop	128
7.8	X server connection	128
7.8.1	X server local connection	128
7.8.2	X server remote connection	129
7.8.3	X server chroot connection	129
7.9	Área de transferência (Clipboard)	129
8	I18N e L10N	131
8.1	O locale	131
8.1.1	Fundamentos para o locale UTF-8	131
8.1.2	A reconfiguração do locale	132
8.1.3	Codificação de nomes de ficheiros	133
8.1.4	Mensagens localizadas e documentação traduzida	133
8.1.5	Efeitos do locale	133
8.2	A entrada do teclado	134
8.2.1	The keyboard input for Linux console and X Window	134
8.2.2	The keyboard input for Wayland	134
8.2.3	O suporte a método de entrada com IBus	134
8.2.4	Um exemplo para Japonês	134
8.3	O ecrã de resultados	135
8.4	Caracteres Asiáticos de Leste de Altura Ambígua	135
9	Dicas do sistema	137
9.1	The console tips	137
9.1.1	Gravar as atividades da shell de modo limpo	137
9.1.2	O programa screen	138
9.1.3	Navigating around directories	138
9.1.4	Readline wrapper	139
9.2	Personalizar o vim	139
9.2.1	Customizing vim with internal features	140
9.2.2	Customizing vim with external packages	140
9.3	Gravação de dados e apresentação	141
9.3.1	O daemon de log	141
9.3.2	Analizador de relatório (Log)	141
9.3.3	Amostragem personalizada de dados em texto	142

9.3.4	Amostragem personalizada de hora e data	142
9.3.5	Echo de shell colorido	142
9.3.6	Comandos coloridos	143
9.3.7	Recordar as atividades do editor para repetições complexas	143
9.3.8	Gravar a imagem gráfica de uma aplicação X	143
9.3.9	Gravar alterações em ficheiros de configuração	144
9.4	Monitorizar, controlar e iniciar as atividades de programas	144
9.4.1	Temporizar um processo	144
9.4.2	A prioridade de agendamento	144
9.4.3	O comando ps	146
9.4.4	O comando top	146
9.4.5	Listar ficheiros abertos por um processo	146
9.4.6	Rastear as atividades de programas	146
9.4.7	Identificação de um processo a usar ficheiros ou sockets	146
9.4.8	Repetir um comando com um intervalo constante	147
9.4.9	Repetir um ciclo de comandos sobre ficheiros	147
9.4.10	Arrancar um programa a partir da GUI	147
9.4.11	Personalizar o programa a ser iniciado	149
9.4.12	Matar um processo	149
9.4.13	Agendar tarefas uma vez	149
9.4.14	Agendar tarefas regularmente	151
9.4.15	Tecla Alt-SysRq	151
9.5	Dicas de manutenção do sistema	152
9.5.1	Quem está no sistema?	152
9.5.2	Avisar todos	152
9.5.3	Identificação do hardware	152
9.5.4	Configuração do hardware	153
9.5.5	Hora do sistema e do hardware	153
9.5.6	A configuração do terminal	154
9.5.7	A infraestrutura de som	154
9.5.8	desativar o protector de ecrã (screensaver)	155
9.5.9	desativar os sons de beep	155
9.5.10	Utilização da memória	155
9.5.11	Segurança do sistema e verificação de integridade	156
9.6	Dicas de armazenamento de dados	157
9.6.1	Utilização do espaço em disco	157
9.6.2	Configuração das partições do disco	157
9.6.3	Aceder a partição a usar UUID	158
9.6.4	LVM2	158

9.6.5	Configuração do sistema de ficheiros	159
9.6.6	Criação do sistema de ficheiros e verificação de integridade	159
9.6.7	Optimização do sistema de ficheiros por opções de montagem	160
9.6.8	Optimização do sistema de ficheiros através do superbloc	160
9.6.9	Optimização do disco rígido	161
9.6.10	Optimização de disco de estado sólido (SSD)	161
9.6.11	Usar SMART para prever falhas no disco rígido	161
9.6.12	Especifique o diretório de armazenamento temporário através de \$TMPDIR	161
9.6.13	Expandir o espaço de armazenamento utilizável via LVM	162
9.6.14	Expandir o espaço de armazenamento utilizável ao montar outra partição	162
9.6.15	Expandir o espaço de armazenamento utilizável ao fazer bind-mount para outro diretório	162
9.6.16	Expansão do espaço de armazenamento utilizável ao fazer overlay-mounting para outro diretório	162
9.6.17	Expandir o espaço de armazenamento utilizável a usar ligações simbólicas	162
9.7	A imagem de disco	163
9.7.1	Criar o ficheiro de imagem de disco	163
9.7.2	Escrever directamente no disco	163
9.7.3	Montar o ficheiro de imagem de disco	164
9.7.4	Limpar um ficheiro de imagem de disco	165
9.7.5	Criar um ficheiro de imagem de disco vazio	165
9.7.6	Criar o ficheiro de imagem ISO9660	166
9.7.7	Escrever directamente ao CD/DVD-R/RW	166
9.7.8	Montar o ficheiro de imagem ISO9660	167
9.8	Os dados binários	167
9.8.1	Ver e editar dados binários	167
9.8.2	Manipular ficheiros sem montar o disco	167
9.8.3	Redundância de dados	168
9.8.4	Recuperação de ficheiros e dados e análise forense	168
9.8.5	Dividir um ficheiro grande em ficheiros pequenos	168
9.8.6	Limpar conteúdo de ficheiro	169
9.8.7	Ficheiros dummy	169
9.8.8	apagar um disco rígido inteiro	169
9.8.9	Apagar uma área não utilizada do disco rígido	170
9.8.10	Recuperar ficheiros apagados mas ainda abertos	170
9.8.11	Procurar todas as ligações rígidas	171
9.8.12	Consumo invisível do espaço do disco	171
9.9	Dicas de encriptação de dados	171
9.9.1	Encriptação de discos amovíveis com dm-crypt/LUKS	172
9.9.2	Partição swap encriptada com dm-crypt	172
9.9.3	Montar discos encriptados com dm-crypt/LUKS	173

9.10	O kernel	173
9.10.1	Parâmetros do kernel	173
9.10.2	Cabeçalhos do kernel	173
9.10.3	Compilar o kernel e módulos relacionados	173
9.10.4	Compilar código-fonte do kernel: a recomendação da equipa do kernel de Debian	174
9.10.5	Controladores de hardware e firmware	175
9.11	Sistema virtualizado	176
9.11.1	Virtualization and emulation tools	176
9.11.2	Fluxo de trabalho da virtualização	176
9.11.3	Montar o ficheiro de imagem de disco virtual	178
9.11.4	Sistema chroot	178
9.11.5	Sistemas de vários ambientes de trabalho	179
10	Gestão de dados	180
10.1	Partilhar, copiar e arquivar	180
10.1.1	Ferramentas de arquivo e compressão	181
10.1.2	Ferramentas de cópia de sincronização	181
10.1.3	Idiomas para o arquivo	181
10.1.4	Idiomas para a cópia	183
10.1.5	Idiomas para a seleção de ficheiros	184
10.1.6	Meio de arquivo	185
10.1.7	Aparelho de armazenamento amovível	186
10.1.8	Escolha de sistema de ficheiros para partilhar dados	187
10.1.9	Partilhar dados via a rede	188
10.2	Salvaguarda (backup) e recuperação	189
10.2.1	Suites de utilitários de backup	190
10.2.2	Personal backup	191
10.3	Infraestrutura da segurança de dados	191
10.3.1	Gestão de chaves para GnuPG	191
10.3.2	Usa GnuPG em ficheiros	193
10.3.3	Usar GnuPG com o Mutt	193
10.3.4	Usar GnuPG com o Vim	194
10.3.5	O valor de controlo MD5	194
10.4	Ferramentas de fusão de código fonte	194
10.4.1	Extraír as diferenças para ficheiros fonte	194
10.4.2	Fundir atualizações para ficheiros de fonte	194
10.4.3	Interactive merge	196
10.5	Git	196
10.5.1	Configuração do cliente Git	196
10.5.2	Basic Git commands	197
10.5.3	Git tips	197
10.5.4	Referências do Git	198
10.5.5	Other version control systems	198

11 Conversão de dados	201
11.1 Ferramentas de conversão de dados em texto	201
11.1.1 Converter um ficheiro de texto com o iconv	202
11.1.2 Verifica ficheiro se é UTF-8 com o iconv	203
11.1.3 Converter os nomes dos ficheiros com o iconv	203
11.1.4 conversão EOL	204
11.1.5 Conversão de TAB	204
11.1.6 Editores com auto-conversão	204
11.1.7 Extracção de texto simples	205
11.1.8 Destacar e formatar dados de texto simples	205
11.2 Dados XML	205
11.2.1 Dicas básicas para XML	207
11.2.2 Processamento de XML	208
11.2.3 A extracção de dados de XML	208
11.2.4 The XML data lint	209
11.3 Formatação de texto	209
11.3.1 formatação de texto roff	210
11.3.2 TeX/LaTeX	210
11.3.3 Impressão bonita de um manual	211
11.3.4 Criar um manual	211
11.4 Dados imprimíveis	211
11.4.1 Ghostscript	211
11.4.2 Juntar dois ficheiros PS ou PDF	212
11.4.3 Utilitários de dados imprimíveis	212
11.4.4 Imprimir com o CUPS	212
11.5 A conversão de dados de mail	212
11.5.1 Noções básicas de dados de mail	214
11.6 Ferramentas de dados gráficos	214
11.7 Conversão de dados variados	216
12 Programação	217
12.1 O script de shell	217
12.1.1 Compatibilidade da shell do POSIX	218
12.1.2 Parâmetros da shell	218
12.1.3 Condicionais da shell	219
12.1.4 Ciclos (loops) da shell	220
12.1.5 A sequência de processamento da linha de comandos da shell	220
12.1.6 Programas utilitários para script de shell	221
12.2 Scripting in interpreted languages	223

12.2.1	Debugging interpreted language codes	223
12.2.2	GUI program with the shell script	223
12.2.3	A loucura dos scripts curtos de Perl	224
12.3	Coding in compiled languages	224
12.3.1	C	225
12.3.2	Programa C simples (gcc)	225
12.3.3	Flex —um Lex melhor	226
12.3.4	Bison —um Yacc melhor	226
12.4	Ferramentas de análise de código estático	227
12.5	Depuração	227
12.5.1	Execução gdb básica	228
12.5.2	Depurar o pacote Debian	228
12.5.3	Obter um backtrace	229
12.5.4	Comandos gdb avançados	229
12.5.5	Verificar a dependência em bibliotecas	229
12.5.6	Dynamic call tracing tools	230
12.5.7	Depurar Erros do X	230
12.5.8	Ferramentas de detecção de fugas de memória	230
12.5.9	Desassemblar binário	231
12.6	Build tools	231
12.6.1	Make	231
12.6.2	Autotools	232
12.6.2.1	Compilar e instalar um programa	232
12.6.2.2	Desinstalar um programa	232
12.6.3	Meson	233
12.7	Web	233
12.8	A tradução do código-fonte	233
12.9	Criar um pacote Debian	234
A	Apêndice	235
A.1	o labirinto Debian	235
A.2	História do Copyright	235
A.3	Formato do documento	236

Lista de Tabelas

1.1	Lista de pacotes de programas interessantes em modo de texto	4
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)	10
1.6	Exemplos do valor umask	11
1.7	Lista de grupos notáveis disponibilizados pelo sistema para acesso a ficheiros	11
1.8	Lista de grupos notáveis disponibilizados pelo sistema para execuções de comandos particulares	12
1.9	Lista dos tipos de marcas temporais	12
1.10	Lista de ficheiros de aparelhos especiais	16
1.11	As teclas de atalho do MC	18
1.12	A reacção à tecla enter no MC	19
1.13	Lista de programas da shell	20
1.14	Lista de teclas de atalho para bash	21
1.15	List of mouse operations and related key actions on Debian	22
1.16	List of basic Vim key strokes	23
1.17	lista dos comandos Unix básicos	25
1.18	As 3 partes do valor locale	26
1.19	Lista de recomendações de locale	27
1.20	Lista de valores "\$HOME"	28
1.21	Padrões glob da shell	28
1.22	Códigos de saída do comando	29
1.23	Idiomas de comandos de shell	30
1.24	Descritores de ficheiro predefinido	31
1.25	Meta-caracteres para BRE e ERE	34
1.26	A expressão de substituição	34
1.27	Lista de trechos de script para canalizar comandos em pipe	37
2.1	Lista de ferramentas de gestão de pacotes Debian	39
2.2	Lista de sites de arquivos Debian	42

2.3	Lista de área de arquivo Debian	42
2.4	A relação entre suite e nome de código	43
2.5	Lista de sites web chave para resolver problemas com um pacote específico	47
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)	49
2.7	Opções de comando notáveis para o aptitude(8)	50
2.8	Lista de teclas de atalho do aptitude	51
2.9	Lista de vistas para o aptitude	52
2.10	A categorização das vista de pacotes standard	52
2.11	Lista da fórmula regex do aptitude	54
2.12	Os ficheiros log para atividades de pacotes	55
2.13	Lista de operações de gestão avançada de pacotes	58
2.14	O conteúdo dos meta dados do arquivo Debian	60
2.15	A estrutura de nomes dos pacotes Debian	63
2.16	Os caracteres utilizáveis para cada componente nos nomes de pacotes Debian	63
2.17	Ficheiros notáveis criados pelo dpkg	64
2.18	Lista de valores notáveis de Pin-Priority para a técnica de apt-pinning .	71
2.19	Lista de ferramentas proxy especiais para arquivos Debian	76
3.1	Lista de gestores de arranque	79
3.2	The meaning of the menu entry of the above part of /boot/grub/grub.cfg	80
3.3	Lista de utilitários de arranque para o sistema Debian	81
3.4	Lista de níveis de erro do kernel	83
3.5	List of typical journalctl command snippets	84
3.6	List of typical systemctl command snippets	85
3.7	List of other monitoring command snippets under systemd	86
4.1	3 ficheiros de configuração importantes para pam_unix(8)	88
4.2	A segunda entrada no conteúdo de "/etc/passwd"	89
4.3	Lista de comandos para gerir informação de conta	90
4.4	Lista de ferramentas para gerar palavras-passe	91
4.5	Lista de sistemas PAM e NSS notáveis	91
4.6	Lista de ficheiros de configuração acedidos pelo PAM e NSS	92
4.7	Lista de serviços e portos inseguros e seguros	94
4.8	Lista de ferramentas para disponibilizar medidas de segurança extra	95
5.1	Lista de ferramentas de configuração de rede	99
5.2	Lista de gamas de endereços de rede	101
5.3	Tabela de tradução dos comandos obsoletos net-tools para os novos comandos iproute2	103
5.4	Lista de comandos de rede de baixo nível	103

5.5	Lista de ferramentas de otimização de rede	104
5.6	Regras básicas para o valor MTU óptimo	105
5.7	Lista de ferramentas de firewall	106
6.1	Lista de exploradores web	107
6.2	Lista de pacotes de plugins de navegadores	108
6.3	Lista de agentes utilizador de mail (MUA)	109
6.4	List of basic mail transport agent related packages	111
6.5	Lista dos manuais importantes do postfix	112
6.6	Lista de ficheiros de configuração relacionados com endereços de mail	113
6.7	Lista de operações MTA básicas	114
6.8	Lista de servidores de acesso remoto e utilitários	115
6.9	Lista de ficheiros de configuração do SSH	116
6.10	Lista de exemplos de arranque do cliente SSH	116
6.11	Lista de clientes SSH para outras plataformas	117
6.12	Lista de servidores de impressoras e utilitários	118
6.13	Lista de outras aplicações de servidor de rede	119
6.14	Lista de clientes de aplicação de rede	120
6.15	Lista de RFCs populares	120
7.1	List of desktop environment	121
7.2	List of notable GUI infrastructure packages	123
7.3	List of notable GUI applications	124
7.4	List of notable TrueType and OpenType fonts	125
7.5	List of notable font environment and related packages	126
7.6	List of notable sandbox environment and related packages	127
7.7	List of notable remote access server	128
7.8	Lista de métodos de ligação ao servidor X	128
7.9	List of programs related to manipulating character clipboard	130
8.1	List of IBus and its engine packages	135
9.1	List of programs to support console activities	137
9.2	Lista de ligações de teclas para o screen	139
9.3	Information on the initialization of vim	141
9.4	Lista de analisadores de log do sistema	141
9.5	Display examples of time and date for the "ls -l" command with the time style value	142
9.6	Lista de ferramentas gráficas de manipulação de imagens	144
9.7	List of packages which can record configuration history	144
9.8	Lista de ferramentas para monitorizar e controlar as atividades de programas	145

9.9	Lista de valores nice para a prioridade de agendamento	145
9.10	Lista dos estilos do comando ps	146
9.11	Lista dos sinais frequentemente usados para o comando kill	150
9.12	List of notable SAK command keys	151
9.13	Lista de ferramenta de identificação de hardware	153
9.14	Lista de ferramentas de configuração do hardware	153
9.15	Lista de pacotes de som	155
9.16	Lista de comandos para desativar o protector de ecrã	155
9.17	Lista dos tamanhos de memória reportados	156
9.18	Lista de ferramentas para segurança do sistema e verificação de integridade	157
9.19	Lista de pacotes de gestão de partições do disco	158
9.20	Lista de pacotes de gestão de sistemas de ficheiros	159
9.21	Lista de pacote para ver e editar dados binários	167
9.22	Lista de pacotes para ler e escrever ficheiros sem montar o disco	167
9.23	Lista de ferramentas para adicionar redundância de dados a ficheiros	168
9.24	Lista de pacotes para recuperação de ficheiros e dados e análise forense	168
9.25	Lista de utilitários de encriptação de dados	171
9.26	Lista de pacotes chave a serem instalados para a recompilação do kernel no sistema Debian	174
9.27	Lista de ferramentas de virtualização	177
10.1	Lista de ferramentas de arquivo e compressão	182
10.2	Lista de ferramentas de cópia e sincronização	183
10.3	Lista de hipóteses de sistemas de ficheiros para aparelhos de armazenamento amovíveis com cenários de utilização típica	187
10.4	Lista de serviços de rede para escolher com o cenário de utilização típico	188
10.5	Lista de suites utilitárias de salvaguarda	190
10.6	Lista de ferramentas de infraestrutura da segurança de dados	192
10.7	Lista de comandos do GNU Privacy Guard para gestão de chaves	192
10.8	Lista do significado do código de confiança	192
10.9	Lista de comandos do GNU Privacy Guard em ficheiros	193
10.10	Lista de ferramentas de fusão de código fonte	195
10.11	Lista de pacotes e comandos relacionados com o git	196
10.12	Main Git commands	198
10.13	Git tips	199
10.14	List of other version control system tools	200
11.1	Lista de ferramentas de conversão de dados em texto	201
11.2	Lista de valores de codificação e a utilização deles	202
11.3	Lista de estilos EOL para diferentes plataformas	204
11.4	Lista de comandos de conversão de TAB dos pacotes <code>bsdmainutils</code> e <code>coreutils</code>	204

11.5	Lista de ferramentas para extracção de dados de texto simples	206
11.6	Lista de ferramentas para destacar dados em texto simples	206
11.7	Lista de entidades predefinidas para XML	207
11.8	Lista de ferramentas XML	208
11.9	Lista de ferramentas DSSSL	208
11.10	Lista de ferramentas de extracção de dados de XML	209
11.11	Lista de ferramentas de impressão bonita de XML	209
11.12	Lista de ferramentas de formatação de texto	209
11.13	Lista de pacotes para ajudar a criar o manual (manpage)	211
11.14	Lista de interpretadores PostScript Ghostscript	211
11.15	Lista de utilitários de dados imprimíveis	213
11.16	Lista de pacotes para ajudar na conversão de dados de mail	213
11.17	Lista de ferramentas de dados gráficos	215
11.18	Lista de ferramentas de conversão de dados variados	216
12.1	Lista dos 'bashisms' típicos	218
12.2	Lista de parâmetros da shell	218
12.3	Lista de expansões de parâmetros de shell	219
12.4	Lista de substituições de parâmetros de shell chave	219
12.5	Lista de operadores de comparação de ficheiros na expressão condicional	220
12.6	Lista de operadores de comparação de cadeias na expressão condicional	220
12.7	Lista de pacotes que contém programas utilitários pequenos para scripts de shell	222
12.8	List of interpreter related packages	222
12.9	List of dialog programs	223
12.10	List of compiler related packages	225
12.11	Lista de geradores de análise LALR compatíveis com Yacc	226
12.12	Lista de ferramentas para análise de código estático	227
12.13	List of debug packages	227
12.14	Lista de comandos gdb avançados	230
12.15	Lista de ferramentas de detecção de fugas de memória	230
12.16	List of build tool packages	231
12.17	Lista de variáveis automáticas do make	232
12.18	Lista de expansões da variável do make	232
12.19	Lista de ferramentas de tradução de código-fonte	234

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 \(version 2.87\)](#) (2021-11-03 08:37:36 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 sistema Debian em si é um alvo em movimento. Isto torna a documentação dele difícil de ser atual e correta. Embora a versão instável atual do sistema Debian tenha sido usada como base para escrever isto, alguns conteúdos podem já estar desatualizados quando 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 softwares pré-compilados de alta qualidade
- Foco em estabilidade e segurança com acesso fácil a atualizações de segurança
- Foco na atualização tranquila para os pacotes de software mais recentes nos arquivos `unstable` e `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. (Princípio **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 [Livro de Mão dos Administradores de Debian](#)
- 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 nome_de_pacote |grep '/man/man.*/'"`
- A **página info** estilo GNU: `"dpkg -L nome_do_pacote |grep '/info/'"`
- Relatórios de bugs http://bugs.debian.org/package_name
- O Debian Wiki em <https://wiki.debian.org/> para os tópicos específicos e em movimento
- Os HOWTOs de The Linux Documentation Project (TLDP) em <http://tldp.org/>
- The Single UNIX Specification de Open Group's The UNIX System Home Page em <http://www.unix.org/>
- A enciclopédia livre Wikipedia em <https://www.wikipedia.org/>

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.

All command examples are run under the English locale "`LANG=en_US.UTF8`". Please don't expect the placeholder strings such as *command-in-root-account* and *command-in-user-account* to be translated in command examples. This is an intentional choice to keep all translated examples to be up-to-date.

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/users-and-` ou pelo seu [URL](https://www.debian.org), 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 2021-08-14 17:04:37 UTC e contém a submissão total de 196036 relatórios sobre 183707 pacotes binários e 26 arquitecturas.

Nota

Por favor note que o arquivo `amd64 unstable` contém atualmente apenas 63210 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 `"/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-Si-` 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-reference` 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
- Mantenha a sua palavra-passe e chaves de segurança seguras
- [KISS \(keep it simple stupid- manté-lo simples estúpido\)](#)
 - Não exagere 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 escreva nada que não compreenda
- Não altere as permissões do ficheiro (antes da revisão de segurança completa)
- Não abandone a shell de root antes de **TESTARES** as tuas alterações
- Tem sempre uma media de arranque alternativa (caneta USB, CD, ...)

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 se enforcar." --- Miquel van Smoorenburg <miquels@cistron.nl>
- "Unix É amigo do utilizador... Apenas é selectivo sobre quem são os seus amigos." --- Tollef Fog Heen <tollef@add.no>

A Wikipedia tem o artigo "[Unix philosophy](#)" 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

Upon starting the system, you are presented with the character based login screen if you did not install any [GUI](#) environment such as [GNOME](#) or [KDE](#) desktop system. Suppose your hostname is `foo`, the login prompt looks as follows.

If you installed a [GUI](#) environment, then you can still get to the character based login prompt by Ctrl-Alt-F3, and you can return to the GUI environment via Ctrl-Alt-F2 (see Secção [1.1.6](#) below for more).

```
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 11 foo tty1

foo login: penguin
Password:
Linux foo 5.10.0-6-amd64 #1 SMP Debian 5.10.28-1 (2021-04-09) 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: Thu May 13 08:47:13 JST 2021 on tty1
foo:~$
```

Está agora na [shell](#). A shell interpreta os seus comandos.

1.1.2 The shell prompt under GUI

If you installed a [GUI](#) environment during the installation, you are presented with the graphical login screen upon starting your system. You type your username and your password to login to the non-privileged user account. Use tab to navigate between username and password, or use the primary click of the mouse.

You can gain the shell prompt under GUI environment by starting a `x-terminal-emulator` program such as `gnome-terminal(1)`, `rxvt(1)` or `xterm(1)`. Under the GNOME Desktop environment, press SUPER-key (Windows-key) and typing in "terminal" to the search prompt does the trick.

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 aparelhos 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

When your desktop menu does not start GUI system administration tools automatically with the appropriate privilege, you can start them from the root shell prompt of the terminal emulator, such as `gnome-terminal(1)`, `rxvt(1)`, or `xterm(1)`. See Secção 1.1.4 and Secção 7.8.



Atenção

Never start the GUI display/session manager under the root account by typing in `root` to the prompt of the display manager such as `gdm3(1)`.

Nunca execute programas GUI remotos que não sejam confiáveis no X Window quando é mostrada informação crítica porque pode "espiar" o seu ecrã X.

1.1.6 Consolas virtuais

Por omissão no sistema Debian existem disponíveis seis consolas de caracteres alternáveis [tipo VT100](#) para arrancar a shell de comandos directamente na máquina Linux. A menos que esteja num ambiente GUI, pode mudar entre consolas virtuais ao pressionar `Left-Alt-key` 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.

If you are in the GUI environment, you gain access to the character console 3 by pressing `Ctrl-Alt-F3` key, i.e., the `left-Ctrl-key`, the `left-Alt-key`, and the `F3-key` are pressed together. You can get back to the GUI environment, normally running on the virtual console 2, by pressing `Alt-F2`.

You can alternatively change to another virtual console, e.g. to the console 3, from the commandline.

```
# chvt 3
```

1.1.7 Como abandonar a linha de comandos

Escreva `Ctrl-D`, isto é, a `tecla-Ctrl-esquerda` e a `tecla d` 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 envolve 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 qualquer-ficheiro-binário`, 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
...
```

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

pacote	popcon	tamanho	descrição
mc	V:55, I:234	1492	Um gestor de ficheiro de ecrã completo em modo de texto
sudo	V:597, I:809	4589	Um programa para permitir privilégios de root limitados aos utilizadores
vim	V:102, I:404	3286	O editor de texto de Unix Vi IMproved, um editor de texto para programadores (versão standard)
vim-tiny	V:57, I:969	1574	O editor de texto de Unix Vi IMproved, um editor de texto para programadores (versão compacta)
emacs-nox	V:4, I:19	18364	Emacs do Projecto GNU, o editor de texto extensível baseado em Lisp
w3m	V:25, I:252	2367	Navegadores de WWW de modo de texto
gpm	V:11, I:16	540	O cortar-e-colar estilo Unix na consola de texto (daemon)

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

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
doc-debian	1:850	166	Documentação do Projecto Debian, (FAQ do Debian) e outros documentos
debian-policy	1:29	4306	Manual de Políticas Debian e documentos relacionados
developers-reference	1:6	1916	Guias e informação para programadores de Debian
debmake-doc	1:0	10989	Guide for Debian Maintainers
debian-history	1:1	4285	História do Projecto Debian
debian-faq	1:846	817	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
```

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 aparelhos 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 aparelhos. `mount(8)` serve para anexar o sistema de ficheiros encontrado num aparelho 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/filesystems/`.

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, `"MEUFICHEIRO"` e `"MeuFicheiro"` 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 **exceto** `"/"`. O diretório raiz é uma exceção. 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/init.d/` 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 aparelho 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 aparelho tal como `"C:\"`. (No entanto, existem entradas nos diretórios que referem-se a aparelhos 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 `"ComoEsteExemplo"`. 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/d` e `hier(7)`). Deve lembrar-se dos seguintes factos como princípio:

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 aparelhos de armazenamento e a interacção com os aparelhos 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 aparelho 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

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

(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 aparelhos 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 ficheiros 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 aparelho de caractere
b	nó de aparelho 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`).

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)`

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 "`rw`" 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:

```
$ 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)
```

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.)

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****Dica**

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

1.2.5 Permissões para grupos de utilizadores (group)

De modo a que as permissões de grupo sejam aplicadas a um utilizador particular, esse utilizador precisa de se tornar membro do grupo a usar "sudo vigr" para `/etc/group` e "sudo vigr -s" para `/etc/gshadow`. Precisa de iniciar sessão após terminar sessão (ou correr "exec newgrp") para ativar a nova configuração do grupo.

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 aparelhos de hardware são apenas outro tipo de ficheiros no sistema Debian. Se tiver problemas a aceder a aparelhos como o CD-ROM e memórias USB 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 aparelhos particulares sem privilégios de root.

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

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	executar sudo sem a palavra-passe deles
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 (<code>ls -l</code>)
ctime	a hora de alteração de estado do ficheiro (<code>ls -lc</code>)
atime	a hora do último acesso ao ficheiro (<code>ls -lu</code>)

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.

For timestamps, the `ls` command outputs localized strings under non-English locale ("`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`".

- [Ligação rígida](#)
 - 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
```

The hardlink can be made within the same filesystem and shares the same inode number which the "`-i`" option with `ls(1)` reveals.

A ligação simbólica tem sempre permissões nominais de acesso ao ficheiro "`rw-rw-rw-`", 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 põe 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

`"netstat -an"` dá uma visão geral, muito útil, dos sockets que estão abertos num determinado sistema.

1.2.10 Ficheiros de aparelho

Os [ficheiros de Aparelhos](#) referem-se a aparelhos físicos ou virtuais no seu sistema, como o seu disco rígido, placa gráfica, monitor ou teclado. Um exemplo de aparelho virtual é a consola, que é representada por `"/dev/console"`.

Existem 2 tipos de ficheiros de aparelho.

- **Aparelho de Caractere**

- Acedido por um caractere de cada vez
- 1 caractere = 1 byte
- Por exemplo, aparelho de teclado, porta serial, ...

- **Aparelho de Bloco**

- acedido em unidades maiores chamadas blocos
- 1 bloco > 1 byte
- Por exemplo, o disco rígido, ...

Pode ler e escrever nos ficheiros de aparelho, 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 aparelho 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 aparelho 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 aparelho 8 e o número menor de aparelho 0. Isto é acessível para leitura e escrita aos utilizadores que pertencem ao grupo `disk`.
- `"/dev/sr0"` tem o número maior de aparelho 11 e o número menor de aparelho 0. Isto é acessível para leitura e escrita aos utilizadores que pertencem ao grupo `cdrom`.
- `"/dev/ttyS0"` tem o número maior de aparelho 4 e o número menor de aparelho 64. Isto é acessível para leitura e escrita aos utilizadores que pertencem ao grupo `dialout`.
- `"/dev/zero"` tem o número 1 no aparelho maior e o número 5 no número de aparelho 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 aparelhos especiais

Existem alguns ficheiros de aparelhos especiais.

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

ficheiro de aparelho	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 aparelhos especiais

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.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-*`).

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 início 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/filesys`) 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
- `/var/log` os seus sub-diretórios
- `/usr/share/doc` os seus sub-diretórios
- `/sbin` e `/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

In some consoles such as `gnome-terminal(1)`, key strokes of function-keys may be stolen by the console program. You can disable these features in "Preferences" → "General" and "Shortcuts" menu for `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 [9.5.6](#).

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 início para ambas janelas como argumentos ao MC; por exemplo, `"mc /etc /root"`.
- `Esc + n-key` → `Fn` (i.e., `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 início 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-Fn` 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. Use 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	reação 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 <code>*.tar.gz</code> e <code>*.deb</code>	explora o conteúdo dele como se fosse um sub-diretório

Tabela 1.12: A reação 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 FTP no MC

O MC pode ser utilizado para aceder a ficheiros pela Internet a usar FTP. Vá ao menu e pressionar F9 e depois escreva `p` para ativar o sistema de ficheiros virtual FTP. Insira um URL no formato `nome-de-utilizador:palavra-passe@nome-de-máquina` o que obtém um diretório remoto que aparece como se fosse local.

Tente `[deb.debian.org/debian]` como URL e explore o arquivo Debian.

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

Pode seleccionar a sua shell de login com `chsh(1)`.

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`.

pacote	popcon	tamanho	Shell do POSIX	descrição
bash	V:796, I:999	6470	Sim	Bash : a GNU Bourne Again SHell (o standard de facto)
bash-completion	V:31, I:922	1523	N/D	programmable completion for the bash shell
dash	V:913, I:993	221	Sim	Shell Almquist , bom para scripts da shell
zsh	V:36, I:74	2467	Sim	Z shell : a shell standard com muitas melhorias
tcsh	V:8, I:28	1316	Não	Shell TENEX C : uma versão melhorada de Berkeley csh
mksh	V:7, I:12	1478	Sim	Uma versão de Korn shell
csh	V:2, I:8	343	Não	OpenBSD shell C , uma versão do Berkeley csh
sash	V:0, I:6	1090	Sim	shell Stand-alone com comandos embutidos (Não se destina a <code>"/bin/sh"</code> standard)
ksh	V:2, I:15	3284	Sim	a real, versão AT&T da Korn shell
rc	V:0, I:1	169	Não	implementação da rc shell AT&T Plan 9
posh	V:0, I:0	190	Sim	Policy-compliant Ordinary SHell (deriva da pdksh)

Tabela 1.13: Lista de programas da shell

1.4.2 Personalizar bash

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

Por exemplo, tente o seguinte.

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

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 Mouse operations

[Mouse operations for text on Debian system mix 2 styles](#) with some twists:

- Traditional Unix style mouse operations:
 - use 3 buttons (click)
 - use PRIMARY
 - used by X applications such as `xterm` and text applications in Linux console
- Modern GUI style mouse operations:
 - use 2 buttons (drag + click)
 - use PRIMARY and CLIPBOARD
 - used in Modern GUI applications such as `gnome-terminal`

Here, the PRIMARY selection is the highlighted text range. Within the terminal program, `Shift-Ctrl-C` is used instead to avoid terminating a running program.

The center wheel on the modern wheel mouse is considered middle mouse button and can be used for middle-click. Clicking left and right mouse buttons together serves as the middle-click under the 2 button mouse system situation.

In order to use a mouse in Linux character consoles, you need to have `gpm(8)` running as daemon.

acção	resposta
Clique-esquerdo-e-arrastar do rato	select range as PRIMARY selection
Clique-esquerdo	select the start of range for PRIMARY selection
Right-click (traditional)	select the end of range for PRIMARY selection
Right-click (modern)	context dependent menu (cut/copy/paste)
Middle-click or Shift-Ins	insert PRIMARY selection at the cursor
Ctrl-X	cut PRIMARY selection to CLIPBOARD
Ctrl-C (Shift-Ctrl-C in terminal)	copy PRIMARY selection to CLIPBOARD
Ctrl-V	paste CLIPBOARD at the cursor

Tabela 1.15: List of mouse operations and related key actions on 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/LESSOPEN`. A opção "-R" permite saída em caracteres em bruto e activa sequências de escape de cores ANSI. Veja `less(1)`.

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 que aprenda 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

Good editors, such as Vim and Emacs, can handle UTF-8 and other exotic encoding texts correctly. It is a good idea to use the GUI environment in the UTF-8 locale and to install required programs and fonts to it. Editors have options to set the file encoding independent of the GUI environment. Please refer to their documentation on multibyte text.

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 Using vim

The recent vim(1) starts itself in the sane "nocompatible" option and enters into the NORMAL mode.¹

mode	key strokes	acção
NORMAL	:help only	display the help file
NORMAL	:e filename.ext	open new buffer to edit filename.ext
NORMAL	:w	overwrite current buffer to the original file
NORMAL	:w filename.ext	write current buffer to filename.ext
NORMAL	:q	quit vim
NORMAL	:q!	force to quit vim
NORMAL	:only	close all other split open windows
NORMAL	:set nocompatible?	check if vim is in the sane nocompatible mode
NORMAL	:set nocompatible	set vim to the sane nocompatible mode
NORMAL	i	enter the INSERT mode
NORMAL	R	enter the REPLACE mode
NORMAL	v	enter the VISUAL mode
NORMAL	V	enter the linewise VISUAL mode
NORMAL	Ctrl-V	enter the blockwise VISUAL mode
except TERMINAL - JOB	ESC-key	enter the NORMAL mode
NORMAL	:term	enter the TERMINAL - JOB mode
TERMINAL - NORMAL	i	enter the TERMINAL - JOB mode
TERMINAL - JOB	Ctrl-W N (or Ctrl-\ Ctrl-N)	enter the TERMINAL - NORMAL mode
TERMINAL - JOB	Ctrl-W :	enter the Ex-mode in TERMINAL - NORMAL mode

Tabela 1.16: List of basic Vim key strokes

Please use the "vimtutor" program to learn vim through an interactive tutorial course.

The vim program changes its behavior to typed key strokes based on **mode**. Typing in key strokes to the buffer is mostly done in INSERT-mode and REPLACE-mode. Moving cursor is mostly done in NORMAL-mode. Interactive selection is done in VISUAL-mode. Typing ":" in NORMAL-mode changes its **mode** to Ex-mode. Ex-mode accepts commands.

Dica

The Vim comes with the **Netrw** package. Netrw supports reading files, writing files, browsing directories over a network, and local browsing! Try Netrw with "vim ." (a period as the argument) and read its manual at ":help netrw".

For the advanced configuration of vim, see 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.

¹Even the older vim can starts in the sane "nocompatible" mode by starting it with the "-N" option.

Dica

The new Vim (version>=8.2) can be used to record the shell activities cleanly using `TERMINAL - JOB-mode`. See 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 exceção. 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.

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 `builtin(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 builtin
$ 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
```

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

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-direções (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.

- The PAM system such as `pam_env` may set environment variables by `/etc/pam.conf`, `/etc/environment` and `/etc/default/locale`.
- The display manager such as `gdm3` may reset environment variables for GUI session.
- The user specific program initialization may reset environment variables by `~/ .profile`, `~/ .bash_profile` and `~/ .bashrc`.

1.5.2 A variável "\$LANG"

The default locale is defined in the "\$LANG" environment variable and is configured as `"LANG=xx_YY.UTF-8"` by the installer or by the subsequent GUI configuration, e.g., "Settings" → "Region & Language" → "Language" / "Formats" for GNOME.

Nota

I recommend you to configure the system environment just by the "\$LANG" variable for now and to stay away from "\$LC_*" variables unless it is absolutely needed.

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

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

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

```
$ 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
```

Here, the program `date(1)` is executed with different locale values.

- 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

When filing a bug report, running and checking the command under “en_US.UTF-8” locale is a good idea if you use non-English environment.

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 omissão, a variável de ambiente “\$PATH” das contas de utilizadores pode não incluir “/sbin” nem “/usr/sbin”. Por exemplo, o comando `ifconfig` necessita ser chamado com o caminho completo como “/sbin/ifconfig”. (De modo idêntico, o comando `ip` está localizado em “/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/utilizador_normal	programa executado a partir da shell de utilizador normal
/home/utilizador_normal	programa executado a partir menu do ambiente GUI do utilizador normal
/home/utilizador_normal	programa executado como root com o "programa sudo"
/root	programa executado como root com o "programa sudo -H"

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/".

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 (segmento) não iniciado por "."
.*	nome de ficheiro (segmento) iniciado por "."
?	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 (a excluir "^")

Tabela 1.21: Padrões glob da shell

Por exemplo, tente o seguinte:

```
$ mkdir junk; cd junk; touch 1.txt 2.txt 3.c 4.h .5.txt ..6.txt
$ echo *.txt
1.txt 2.txt
$ echo *
1.txt 2.txt 3.c 4.h
$ echo *. [hc]
3.c 4.h
$ echo .*
. . . .5.txt ..6.txt
$ echo .*[^\.]*
.5.txt ..6.txt
$ echo [^1-3]*
4.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 redirecionamento da shell

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

idioma do comando	descrição
comando &	execução em segundo plano do comando na sub-shell
comando1 comando2	liga em pipe a saída standard do comando1 à entrada standard do comando2 (execução concorrente)
comando1 2>&1 comando2	liga em pipe a saídas standard e o erro standard do comando1 à entrada standard do comando2 (execução concorrente)
comando1 ; comando2	executa o comando1 e o comando2 sequencialmente
comando1 && comando2	executa o comando1; se tiver sucesso, executa o comando2 sequencialmente (retorna sucesso se ambos comando1 e comando2 tiverem sucesso)
comando1 comando2	executa o comando1; se não tiver sucesso, executa o comando2 sequencialmente (retorna sucesso se o comando1 ou o comando2 tiverem sucesso)
comando > foo	redirecciona a saída standard do comando para o ficheiro foo (sobrescreve)
comando 2> foo	redirecciona o erro standard do comando para o ficheiro foo (sobrescreve)
comando >> foo	redirecciona a saída standard do comando para o ficheiro foo (acrescenta)
comando 2>> foo	redirecciona o erro standard do comando ao ficheiro foo (acrescenta)
comando > foo 2>&1	redirecciona ambos saída standard e erro standard do comando para o ficheiro foo
comando < foo	redirecciona a entrada standard do comando ao ficheiro foo
comando << delimitador	redirecciona a entrada standard do comando para as seguintes linhas até que o "delimitador" seja atingido (documentar aqui)
comando <<- delimitador	redirecciona a entrada standard do comando às seguintes linhas até que o "delimitador" seja atingido (aqui documento, os caracteres tab de inicio são retirados das linhas de entrada)

Tabela 1.23: Idiomas de comandos 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
```

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 3foo 4bar # 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.

aparelho	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.
- Basic regular expression (**BRE**) is used as default:
 - `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.
 - `tc l(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)`.
- `pcgrep(1)` do pacote `pcgrep` 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/html/`

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

Nota

Sort order and range expression are locale dependent. If you wish to obtain traditional behavior for a command, use **C** locale or **C.UTF-8** locale instead of normal **UTF-8** ones (see 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.

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.

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 's/\(1[a-z]*\)[0-9]*\(.*\)$/\2===\1/'
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
```

BRE	ERE	descrição da expressão regular
\ . [] ^ \$ *	\ . [] ^ \$ *	meta-caracteres comuns
\+ \? \ (\) \{ \} \		BRE apenas meta-caracteres 'escapados' "\
	+ ? () { }	ERE apenas meta-caracteres não 'escapados' "\
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

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

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

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

[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](#).

The current Debian package management system which can utilize all these resources is [Advanced Packaging Tool \(APT\)](#).

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 63210 pacotes disponíveis para a arquitetura 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)` for all interactive command line operations, including package installation, removal and dist-upgrades.
- `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.

2.1 Pré-requisitos da gestão de pacotes Debian

2.1.1 Configuração de pacotes

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

- 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.
-

pacote	popcon	tamanho	descrição
dpkg	V:932, I:999	6874	low level package management system for Debian (file based)
apt	V:887, I:999	4337	APT front-end to manage packages with CLI: apt / apt-get / apt-cache
aptitude	V:72, I:398	4256	APT front-end to interactively manage packages with full screen console: aptitude (8)
tasksel	V:33, I:975	395	APT front-end to install selected tasks: tasksel (8)
unattended-upgrades	V:326, I:447	326	pacote de melhoria para o APT para ativar a instalação automática de atualizações de segurança
gnome-software	V:112, I:203	6559	Software Center for GNOME (GUI APT front-end)
synaptic	V:39, I:308	7873	graphical package manager (GTK APT front-end)
apt-utils	V:379, I:997	1164	Programas utilitários do APT: apt-extracttemplates (1), apt-ftpparchive (1) e apt-sortpkgs (1)
apt-listchanges	V:356, I:849	423	ferramenta de notificação do histórico de alterações do pacote
apt-listbugs	V:8, I:12	465	lista bugs críticos antes de cada instalação do APT
apt-file	V:17, I:79	90	Utilitário de busca de pacotes do APT - interface de linha de comandos
apt-rdepends	V:0, I:6	40	lista recursivamente dependências de pacotes

Tabela 2.1: Lista de ferramentas de gestão de pacotes Debian

- 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.

2.1.2 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 [novato](#) de sistemas Debian deve manter-se com o lançamento **stable** de Debian e aplicar apenas as atualizações de segurança. Quero dizer que é melhor evitar algumas das seguintes acções válidas, como uma precaução, até que compreenda muito bem o sistema Debian. Aqui ficam algumas lembranças.

- Não inclua **testing** ou **unstable** em `"/etc/apt/sources.list"`.
- Não misture Debian standard com outros arquivos que não sejam Debian, como o Ubuntu em `"/etc/apt/sources.list"`.
- 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 ao sistema de gestão de pacotes de Debian, pelas acções referidas acima, podem deixar o seu sistema inutilizado.

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.

2.1.3 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.

Despite my warnings above, I know many readers of this document may wish to run the newer `testing` or `unstable` suites. 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.

This list is targeted for the **self-administered** Desktop environment.

- Use the `testing` suite since it is practically the rolling release automatically managed by the Debian archive QA infrastructure such as the [Debian continuous integration](#), the [source only upload practices](#), and the [library transition tracking](#). The packages in the `testing` suite are updated frequently enough to offer all the latest features.
- Set the codename corresponding to the `testing` suite (currently `"bookworm"`) in the `"/etc/apt/sources.list"`.
- Manually update this codename in the `"/etc/apt/sources.list"` to the new one only after assessing situation by yourself for about a month after the major suite release. The Debian user and developer mailing list are good sources of information for this, too.

The use of the `unstable` suite isn't recommended. The `unstable` suite is **good for debugging packages** as a developer but tends to expose you to unnecessary risks for the normal Desktop usage. Even though the `unstable` suite of the Debian system looks very stable for most of the times, there have been some package problems and a few of them were not so trivial to resolve.

Here are some basic precautionary measure ideas to ensure quick and easy recovery from bugs in Debian packages.

- Faça um sistema de **duplo arranque** ao instalar a suite `stable` do sistema Debian noutra partição
- Tenha o CD de instalação 'à mão' para o **arranque de recuperação**
- 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
- Install a corresponding sandboxed upstream binary package in case of trouble (see Secção [7.6](#))
- Crie um chroot ou ambiente semelhante para antecipadamente correr nele o sistema mais recente (veja Secção [9.11](#))



Cuidado

If you can not do any one of these precautionary actions, you are probably not ready for the `testing` and `unstable` suites.

2.1.4 Básico do arquivos Debian

Vamos olhar ao [arquivo Debian](#) a partir da perspectiva do utilizador do sistema.

Dica

A política oficial do arquivo Debian está definida em [Manual de Políticas Debian, Capítulo 2 - O Arquivo Debian](#).

Para o típico acesso HTTP, o arquivo está especificado no ficheiro `/etc/apt/sources.list` como o seguinte exemplo para o sistema `stable = bullseye` atual.

```
deb http://deb.debian.org/debian/ bullseye main contrib non-free
deb-src http://deb.debian.org/debian/ bullseye main contrib non-free

deb http://security.debian.org/debian-security bullseye-security main contrib
deb-src http://security.debian.org/debian-security bullseye-security main contrib
```

Aqui uso o nome de código `"bullseye"` em vez do nome de suite `"stable"` para evitar surpresas quando a próxima `stable` for lançada.

O significado de `/etc/apt/sources.list` é descrito em `sources.list(5)` e os pontos chave são os seguintes:

- A linha `"deb"` define os pacotes binários.
- A linha `"deb-src"` define os pacotes fonte.
- O 1º argumento é o URL raiz do arquivo Debian.
- O 2º argumento é o nome da distribuição: seja o nome de suite ou o nome de código.
- O 3º argumento e seguintes são a lista de nomes de área de arquivo válidos do arquivo Debian.

As linhas `"deb-src"` pode ser omitidas (ou comentadas ao pôr um `"#"` no início da linha) se for apenas para o `aptitude` o qual não acede a meta-dados relacionados com a fonte. Isso acelera as atualizações dos meta-dados do arquivo. O URL pode ser `"http://"`, `"ftp://"`, `"file://"`, ...

Dica

Se for usado `"sid"` no exemplo em cima em vez de `"bullseye"`, a linha `deb: http://security.debian.org/ ...` para atualizações de segurança em `/etc/apt/sources.list`, não é necessária. Isto porque não há arquivo de atualizações de segurança para `"sid"` (`unstable`).

Aqui está uma lista de URLs de sites de arquivo Debian e nomes das suites ou nomes de código utilizados no ficheiro de configuração.

Cuidado



Apenas o puro lançamento **stable** com as atualizações de segurança disponibilizam a melhor estabilidade. Correr o lançamento **stable** misturado com alguns pacotes dos lançamentos **testing** ou **unstable** é mais arriscado que correr o lançamento **unstable** puro devido a versões erradas de bibliotecas e etc. Se realmente precisa da versão mais recente de alguns programas sob o lançamento **stable**, por favor utilize pacotes do [bullseye-updates](#) e <http://backports.debian.org> (veja os serviços Secção 2.7.4) Estes serviços têm de ser utilizados com cuidados 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 2.7.3).

URL do arquivo	nome da suite (nome de código)	objetivo
http://deb.debian.org/debian/	stable (bullseye)	lançamento (bullseye) stable
http://deb.debian.org/debian/	testing (bookworm)	lançamento (bookworm) testing
http://deb.debian.org/debian/	unstable (sid)	lançamento (sid) unstable
http://deb.debian.org/debian/	experimental	pré-lançamento experimental (opcional, apenas para programadores)
http://deb.debian.org/debian/	stable-proposed-updates (bullseye-proposed-updates)	Updates for the next stable (bullseye) point release (optional)
http://deb.debian.org/debian/	stable-updates (bullseye-updates)	compatible updates for spam filter, IM clients, etc. for stable (bullseye)
http://deb.debian.org/debian/	stable-backports (bullseye-backports)	newer backported packages for stable (bullseye) (optional)
http://security.debian.org/debian-security/	stable-security (bullseye-security)	security updates for stable release (bullseye) (important)
http://security.debian.org/debian-security/	testing-security (bookworm-security)	security updates for testing release (This isn't actively supported by the security team)

Tabela 2.2: Lista de sites de arquivos Debian

Dica

For the Debian system with the stable suite, it is a good idea to include lines with "http://security.debian.org/" in the "/etc/apt/sources.list" to enable security updates as in the example above.

Nota

Os bugs de segurança do arquivo stable são corrigidos pela equipa de segurança do Debian. Esta atividade tem sido bastante rigorosa e fidedigna. Os do arquivo testing poderão ser corrigidos pela equipa de segurança de Debian testing. Por [várias razões](#), esta atividade não é tão rigorosa como a de stable e pode necessitar de aguardar pela migração de pacotes unstable com as correcções. Os pacotes do arquivo unstable são corrigidos pelo maintainer. Os pacotes unstable mantidos activamente estão geralmente em boa forma por conterem as correcções de segurança mais recentes desde a origem. Veja a [FAQ de segurança Debian](#) para saber como Debian lida com os bugs de segurança.

área	quantidade de pacotes	critério do componente do pacote
main	62089	em conformidade com DFSG e nenhuma dependência a non-free
contrib	351	em conformidade com DFSG mas com dependências a non-free
non-free	770	não complacente com DFSG

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 2.1.5).

A organização do arquivo Debian pode ser melhor estudada ao apontar o seu navegador a cada URL de arquivo seguido de `dist`s 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?](#)

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".)

Tempo	suite = stable	suite = testing	suite = unstable
após o lançamento bullseye	nome de código = bullseye	nome de código = bookworm	nome de código = sid
após o lançamento bookworm	nome de código = bookworm	nome de código = trixie	nome de código = sid

Tabela 2.4: A relação entre suite e nome de código

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 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, o arquivo **unstable** pode ser usado 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 correção de bug.

Veja [Manual de Políticas Debian](#) para as definições do arquivo.

- "Secções"

- ["Prioridades"](#)
- ["Sistema base"](#)
- ["Pacotes essenciais"](#)

2.1.5 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 no `main` depende ou recomenda pacotes do `non-free` ou do `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 irão alojar alguns pacotes `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`.
- Os pacotes fora do sistema Debian são alojado em servidores Debian nas áreas `non-free` 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
 - Reconhecemos que alguns dos nossos utilizadores necessitam utilizar trabalhos que não estão de acordo com a Debian Free Software Guidelines. Criamos as áreas `"contrib"` e `"non-free"` no nosso arquivo para esses trabalhos. Os pacotes nessas áreas não fazem parte do sistema Debian, embora tenham sido configurados para serem utilizados com Debian. Encorajamos os fabricantes de CDs a ler as licenças dos pacotes nessas áreas e determinar se podem distribuir os pacotes nos seus CDs. Por isso, embora os trabalhos não-livres não sejam parte de Debian, suportamos a utilização deles e disponibilizamos infraestrutura para os pacotes não-livres (tal como o nosso sistema de seguimento de bugs e listas de email).

Os utilizadores devem estar cientes dos riscos da utilização de pacotes das áreas `non-free` 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` 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 a conter dados binários sem código-fonte tais como os listados em Secção 9.10.5 como não-livre. (a maioria encontra-se na secção `non-free/kernel`.)
- Pacotes de fontes e jogos com restrições comerciais de utilização e/ou modificação de conteúdo.

Por favor note que a quantidade de pacotes das `non-free` e `contrib` é menos de 2% dos pacotes da `main`. ativar o acesso às áreas `non-free` 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.

2.1.6 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:

- "Depends"
 - Isto declara uma dependência absoluta e todos os pacotes listados neste campo têm de ser instalados ao mesmo tempo ou com antecedência.
- "Pre-Depends"
 - Isto é como o Depends, excepto que requer a instalação completa de todos os pacotes listados com antecedência.
- "Recommends"
 - 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.
- "Suggests"
 - 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.
- "Enhances"
 - Isto declara uma dependência fraca como o Suggests mas funciona na direcção oposta.
- "Breaks"
 - Isto declara uma incompatibilidade do pacote normalmente com alguma especificação de versão. Geralmente a resolução é atualizar todos os pacotes listados neste campo.
- "Conflicts"
 - Isto declara uma incompatibilidade absoluta. Todos os pacotes listados neste campo têm de ser removidos para instalar este pacote.
- "Replaces"
 - Isto é declarado quando os ficheiros instalados por este pacote substituem ficheiros nos pacotes listados.
- "Provides"

- 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](#).

2.1.7 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 2.7.3](#) 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.

2.1.8 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_name/README.Debian`. 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)	<code>sensible-browser "http://bugs.debian.org/"</code>
O relatório de bug de um nome de pacote conhecido	<code>sensible-browser "http://bugs.debian.org/nome_do_pacote"</code>
O relatório de bug de uma quantidade de bugs conhecida	<code>sensible-browser "http://bugs.debian.org/número_do_bug"</code>

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.debian.org`, etc.

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

2.2 Operações básicas de gestão de pacotes

As operações de gestão de pacotes baseadas em repositório no sistema Debian podem ser executadas 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 contra o 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 seleccionar "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 resolvidor 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 invólucro 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 uma 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.
-

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
N/D	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)

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` and `aptitude` can be mixed without major troubles.

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 resolvidor 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`".

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)`

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 **ações pendentes** como se fosse arrancado interativamente. Se não gostar disso, pode redefinir isto a partir do menu: "Acção" → "Cancelar ações pendentes".

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)
U	coloca o pacote em retenção (hold)
g	marca todos os pacotes com atualizações (funciona como full-upgrade)
q	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

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:

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    libsmblclient    -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 "ação planeada" (a segunda letra)
- A flag "automático" (a terceira letra)
- O nome do Pacote
- A alteração na utilização do espaço do disco atribuída a "ação 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 2.7.3).
Estão disponíveis vários tipos de vistas de pacotes sob o menu "Vistas".

vista	descrição da vista
Vista de Pacote	veja Tabela 2.10 (predefinição)
Recomendações de Auditoria	lista pacotes que são recomendados por alguns pacotes instalados mas ainda não estão instalados
Lista de Pacotes Lisa	lista pacotes sem categorização (para utilizar com expressões regulares)
Explorador de Debtags	lista pacotes categorizados de acordo com as suas entradas debtags
Source Package View	list packages grouped by source packages

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
Pacotes atualizáveis	lista pacotes organizados como secção → área → pacote
Pacotes Novos	, ,
Pacotes Instalados	, ,
Pacotes Não Instalados	, ,
Pacotes Obsoletos ou Criados Localmente	, ,
Pacotes Virtuais	lista pacotes com a mesma função
Tarefas	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 especifica 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.

- `"~Pterm" == "~Dprovides:term"`
- `"~Cterm" == "~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**.

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.

descrição da regra de correspondência extensa	fórmula da expressão regular
corresponde com o nome do pacote	<code>~nregex_name</code>
corresponde com a descrição	<code>~dregex_description</code>
corresponde com nome da tarefa	<code>~tregex_task</code>
corresponde com debtag	<code>~Gregex_debtag</code>
corresponde com o maintainer	<code>~mregex_maintainer</code>
corresponde com secção do pacote	<code>~sregex_section</code>
corresponde com versão do pacote	<code>~Vregex_version</code>
corresponde com arquivo	<code>~A{bullseye,bookworm,sid}</code>
corresponde com origem	<code>~O{debian,...}</code>
prioridade da correspondência	<code>~p{extra,important,optional,required,standard}</code>
corresponde com pacotes essenciais	<code>~E</code>
corresponde com pacotes virtuais	<code>~V</code>
corresponde com pacotes novos	<code>~N</code>
corresponde com acções pendentes	<code>~a{install,upgrade,downgrade,remove,purge,hold,keep}</code>
corresponde com os pacotes instalados	<code>~i</code>
corresponde com pacotes instalados com marca A (pacotes instalados automaticamente)	<code>~M</code>
corresponde com pacotes instalados sem a marca A (pacotes selecionados pelo administrador)	<code>~i!~M</code>
corresponde com pacotes instalados e com atualizações disponíveis	<code>~U</code>
corresponde com pacotes removidos mas não purgados	<code>~c</code>
corresponde com pacotes removidos, purgados ou que podem-ser-removidos	<code>~g</code>
corresponde com pacotes que declaram dependências quebradas	<code>~b</code>
corresponde com pacotes que declaram dependências quebradas de <i>type</i>	<code>~Btype</code>
corresponde a pacotes <i>pattern</i> que declaram dependência de <i>type</i>	<code>~D[type:]pattern</code>
corresponde a pacotes <i>pattern</i> que declaram dependência quebrada de <i>type</i>	<code>~DB[type:]pattern</code>
corresponde a pacotes para os quais o pacote que corresponde a <i>pattern</i> declara o <i>type</i> de dependência	<code>~R[type:]pattern</code>
corresponde a pacotes para os quais o pacote que corresponde a <i>pattern</i> declara o <i>type</i> de dependência quebrada	<code>~RB[type:]pattern</code>
corresponde com pacotes com os quais alguns pacotes instalados dependem	<code>~R~i</code>
corresponde com pacotes com os quais nenhum outro pacote instalado depende	<code>!~R~i</code>
corresponde com pacotes com os quais alguns pacotes instalados dependem ou recomendam	<code>~R~i ~Rrecommends:~i</code>
corresponde o pacote <i>pattern</i> com a versão filtrada	<code>~S filter pattern</code>
corresponde com todos os pacotes (true)	<code>~T</code>
não corresponde com nenhum pacote (false)	<code>~F</code>

Tabela 2.11: Lista da fórmula regex do aptitude

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 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.

2.3.2 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.

2.3.3 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.

2.3.4 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 "~prequired|~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.

2.3.5 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 colecionados 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 executar a atualização geral do sistema para um lançamento mais recente ao alterar o conteúdo do ficheiro `/etc/apt/sources.` a apontar a um lançamento novo e a executar o comando `apt update; apt dist-upgrade`.

Para atualizar de `stable` para `testing` ou `unstable`, substitui "bullseye" no exemplo `/etc/apt/sources.list` de Secção 2.1.4 por "bookworm" 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 das tarefas de ambiente de trabalho.
8. Remova o ficheiro `/etc/apt/preferences` (desactiva o apt-pinning).
9. Tente a atualização em passos inteligentes: `oldstable` → `stable` → `testing` → `unstable`.
10. atualize o ficheiro `/etc/apt/sources.list` para apontar apenas ao novo arquivo e corra `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.

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`.

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

**Cuidado**

As ferramentas de pacotes de nível mais baixo como `"dpkg -i ..."` e `"debinstall ..."` 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"` necessitam de `"deb-src"` em `"/etc/apt/sources.list"`.
- 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.

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/status"` 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 Salvar para problemas de pacotes

Muito utilizadores preferem seguir o lançamento **unstable** do sistema Debian pelas suas novas funcionalidades e pacotes. Isto torna o sistema permeável a 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 coincidente. 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_exp *"` 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.

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/arquitetura	descrição do arquivo utilizada para a regra do <code>apt_preferences(5)</code>
Packages	topo de cada combinação de distribuição/área/arquitetura-binário	<code>debian/control</code> concatenado para pacotes binários
Sources	topo de cada combinação de distribuição/área/fonte	<code>debian/control</code> concatenado para pacotes fonte

Tabela 2.14: O conteúdo dos meta dados do arquivo Debian

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

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/dists/unstable>" 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 2.1.4. 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 "Release" de nível de topo é verificada pela infraestrutura criptográfica chamada [secure apt](#).

- 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.
- A chave do arquivo Debian público pode ser semeada em "/etc/apt/trusted.gpg";
 - automaticamente ao instalar o chaveiro com o pacote `base-files` mais recente, ou
 - manualmente pela ferramenta `gpg` ou `apt-key` com [a chave do arquivo público mais recente publicada em ftp-master.debian.org](#).
- O sistema **secure APT** verifica a integridade do ficheiro "Release" de nível de topo descarregado criptograficamente por este ficheiro "Release.gpg" a pela chave de arquivo público Debian em "/etc/apt/trusted.gpg".

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).

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" do nível de arquivo para todas as localizações do arquivo especificadas pela linha "deb" em "/etc/apt/sources.list", tais como "<http://deb.debian.org/debian/dists/unstable/main/binary-amd64>" ou "<http://deb.debian.org/debian/dists/sid/main/binary-amd64/Release>" conforme a seguir:

```
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 `bullseye-backports`, que contêm pacotes que não devem ser instalados automaticamente, existe uma linha extra, p.e., "`http://deb.debian.org/debian/dists/experimental/main/binary`" 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 2.7.3).

2.5.4 Obter os meta dados do pacote

Quando as ferramentas do APT, como o `aptitude`, `apt-get`, `synaptic`, `apt-file`, `auto-apt`, ... são utilizadas, 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 aos nomes de distribuição, área e arquitetura especificados em "`/etc/apt/sources.list`" (veja Secção 2.1.4).

- "`/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-arquitetura_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-arquitetura.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 pelo "`apt-get update`" ou "`aptitude update`". Os meta-dados "`Packages`" são atualizados se existir a linha "`deb`" em "`/etc/apt/sources.list`". Os meta dados "`Sources`" são atualizados se existir a linha "`deb-src`" em "`/etc/apt/sources.list`".

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-arquitetura`" 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-arquitetura.gz`" por predefinição.)

2.5.5 O estado dos pacotes 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 Cópia 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 política 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 <code>deb</code>)	<i>nome_de_pacote_versão_upstream-versão-debian_arquitetura</i>
O pacote binário para <code>debian-installer</code> (a.k.a <code>udeb</code>)	<i>nome_de_pacote_versão_upstream-versão-debian_arquitetura</i>
O pacote fonte (código-fonte atual)	<i>nome_de_pacote_versão_upstream-versão-debian.orig.tar.gz</i>
O pacote de código-fonte 1.0 (alterações do Debian)	<i>nome_de_pacote_versão_upstream-versão-debian.diff.gz</i>
O pacote de código-fonte 3.0 (<code>quilt</code>) (alterações do Debian)	<i>nome_de_pacote_versão_upstream-versão-debian.debian.tar.gz</i>
O pacote de código-fonte (descrição)	<i>nome_do_pacote_versão_upstream-versão-debian.dsc</i>

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	usable characters (ERE regex)	existência
<i>nome-do-pacote</i>	<code>[a-z0-9] [-a-z0-9.]+</code>	necessário
<i>epoch:</i>	<code>[0-9]+:</code>	opcional
<i>versão-upstream</i>	<code>[-a-zA-Z0-9.+:]+</code>	necessário
<i>versão.debian</i>	<code>[a-zA-Z0-9.+\~]+</code>	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 *"nome_de_pacote"*, 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
<code>/var/lib/dpkg/info/nome_do_pacote.conf</code>	os ficheiros de configuração. (modificável pelo utilizador)
<code>/var/lib/dpkg/info/nome_do_pacote.list</code>	lista de ficheiros e directórios instalados pelo pacote
<code>/var/lib/dpkg/info/nome_do_pacote.md5sums</code>	lista de sumas de hash MD5 para os ficheiros instalados pelo pacote
<code>/var/lib/dpkg/info/nome_do_pacote.preinst</code>	script do pacote para ser executado antes da instalação do pacote
<code>/var/lib/dpkg/info/nome_do_pacote.postinst</code>	script do pacote para ser executado após a instalação do pacote
<code>/var/lib/dpkg/info/nome_do_pacote.prerm</code>	script do pacote para ser executado antes da remoção do pacote
<code>/var/lib/dpkg/info/nome_do_pacote.postrm</code>	script do pacote para ser executado após a remoção do pacote
<code>/var/lib/dpkg/info/nome_do_pacote.templates</code>	cópias do pacote para o sistema <code>debconf</code>
<code>/var/lib/dpkg/alternatives/nome_do_pacote</code>	a informação alternativa usada pelo comando <code>update-alternatives</code>
<code>/var/lib/dpkg/available</code>	a informação de disponibilidade para todos os pacotes
<code>/var/lib/dpkg/diversions</code>	a informação de diversões usadas pelo <code>dpkg(1)</code> e definidas por <code>dpkg-divert(8)</code>
<code>/var/lib/dpkg/statoverride</code>	a informação de sobreposição de estado usada pelo <code>dpkg(1)</code> e definida por <code>dpkg-statoverride(8)</code>
<code>/var/lib/dpkg/status</code>	a informação de estado para todos os pacotes
<code>/var/lib/dpkg/status-old</code>	o backup de primeira geração do ficheiro <code>"var/lib/dpkg/status"</code>
<code>/var/backups/dpkg.status*</code>	o backup de segunda geração e os mais antigos do ficheiro <code>"var/lib/dpkg/status"</code>

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`. Por exemplo, pode fazer o comando `vi` seleccionar 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 mantém 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, aparelhos, 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 `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 significativa 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 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 2.1.6).

Erros do responsável do pacote ou de implantação inconsistente de mistura de fontes de arquivos (veja Secção 2.7.2) 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
```

2.6.3 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 do script de pacote de "*nome_do_pacote*", 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"`

Configurar todos os pacotes parcialmente instalados com o seguinte comando.

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

2.6.4 Recuperação com o comando dpkg

Como o `dpkg` é uma ferramenta de pacotes de muito baixo nível, pode funcionar sob situações muito más como um sistema que não arranca sem ligação a rede. Vamos assumir que o pacote `foo` está danificado e precisa de ser substituído.

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
```

Dica

Se os danos no sistema forem menores, pode em alternativa fazer um downgrade (regredir a versão) ao sistema completo como em Secção 2.7.10 a usar o nível mais alto do sistema APT.

Se o seu sistema não puder arrancar pelo disco rígido, precisa procurar outras maneiras de arrancá-lo.

1. Arranque o sistema a usar o CD de instalação de Debian (`debian-installer`) em modo de recuperação.
2. Monte o sistema danificado no disco rígido em `"/target"`.
3. Instale uma versão antiga do pacote `foo` com o seguinte.

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

Este exemplo funciona mesmo se o comando `dpkg` no disco rígido estiver danificado.

Dica

Pode ser utilizado, de modo semelhante, para recuperar um sistema danificado qualquer sistema GNU/Linux arrancado de outro sistema no disco rígido, Live CD de GNU/Linux, por pen USB de arranque ou arranque pela rede.

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 a `"--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.

2.6.5 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 serão apresentados nomes de pacotes para instalar. (Podem existir alguns nomes que não são pacotes como `"texmf"`.)

2.7 Dicas para a gestão de pacotes

2.7.1 Como escolher os pacotes Debian

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

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
- Area: main > contrib > non-free
- Prioridade: required > important > standard > optional > extra
- Tasks: pacotes listados em tarefas como "Ambiente de Trabalho"
- Packages selected by the dependency package (e.g., `gcc-10` by `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.

2.7.2 Pacotes de fontes de arquivos misturados



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 [bullseye-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 cria o ficheiro `"/etc/apt/preferences"` nem precisa de se preocupar com o apt-pinning com esta abordagem manual. Mas é muito incómoda.

**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 seguro** 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

Excepto para evitar pacotes com problemas a curto prazo, instalar pacotes binários de arquivos não suportados oficialmente é geralmente uma má ideia. Isto é verdadeiro mesmo que utilize apt-pinning (veja Secção [2.7.3](#)). Deve considerar o chroot ou técnicas semelhantes (veja Secção [9.11](#)) para correr programas de arquivos diferentes.

2.7.3 Moldar a versão candidata



Atenção

A utilização de apt-pinning por um utilizador novato é certamente pedir grandes problemas. Deve evitar utilizar o apt-pinning excepto quando necessitar absolutamente dele.

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 2.7.2), 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**.



Cuidado

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



Cuidado

Os ficheiros Release de nível de arquivo (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`.



Cuidado

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 2.6.4 e Secção 2.7.2) 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 2.7.10).

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 do **lançamento alvo** pode ser definido por diferentes métodos.

Pin-Priority	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**.

- ficheiro de configuração `/etc/apt/apt.conf` com a linha `APT::Default-Release "stable";`
- opção de linha de comandos, p.e. `apt-get install -t testing algum-pacote`

Os arquivos **NotAutomatic** e **ButAutomaticUpgrades** são definidos pelo servidor de arquivo que contém 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`.

- 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/ bullseye-backports/main Packages`.

2.7.4 atualizações e Backports

Existem os arquivos [bullseye-updates](#) e [backports.debian.org](#) que disponibilizam pacotes de atualização para `stable` (bullseye).

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/ bullseye main contrib non-free
deb http://security.debian.org/ bullseye-security main contrib
deb http://deb.debian.org/debian/ bullseye-updates main contrib non-free
deb http://deb.debian.org/debian/ bullseye-backports main 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 `bullseye-updates`.
- Apenas os pacotes antigos instalados manualmente a partir de `bullseye-backports` são atualizados para mais recentes a partir de `bullseye-backports`.

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

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

2.7.5 Bloquear pacotes instalados por "Recomendados"



Atenção

A utilização de apt-pinning por um utilizador novato é certamente pedir grandes problemas. Deve evitar utilizar o apt-pinning excepto quando necessitar absolutamente dele.

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
```

2.7.6 Acompanhar testing com alguns pacotes de unstable



Atenção

A utilização de apt-pinning por um utilizador novato é certamente pedir grandes problemas. Deve evitar utilizar o apt-pinning excepto quando necessitar absolutamente dele.

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 de segue o `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/ 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 arquivo `testing` a usar o arquivo `testing` atual e os pacotes que foram instalados a partir do arquivo `unstable` a usar o arquivo `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/preferences`.

```
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`.

2.7.7 Acompanhar `unstable` com alguns pacotes de experimental

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 ($\ll 100$) 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.

2.7.8 Descarga e atualização automática de pacotes



Atenção

A utilização de apt-pinning por um utilizador novato é certamente pedir grandes problemas. Deve evitar utilizar o apt-pinning excepto quando necessitar absolutamente dele.

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-upgrades/README`.

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 estiver a correr um sistema `unstable`, não vai utilizar as atualizações automáticas, pois um dia, com certeza, irão danificar o seu sistema. Mesmo para casos de `unstable`, pode ainda querer descarregar os pacotes com antecedência para poupar tempo na atualização interativa com parâmetros de configuração como os a seguir.

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

2.7.9 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";
```

2.7.10 Downgrade de emergência



Atenção

A utilização de apt-pinning por um utilizador novato é certamente pedir grandes problemas. Deve evitar utilizar o apt-pinning excepto quando necessitar absolutamente dele.



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 2.7.3). Esta é uma alternativa preguiçosa às acções tediosas de muitos comandos `dpkg -i pacote-danificado-versão-antiga.deb` (veja Secção 2.6.4).

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/ bookworm 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.

2.7.11 Quem fez o upload do pacote?

Apesar do nome do responsável listado em `/var/lib/dpkg/available` e `/usr/share/doc/package_name/changelog` 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.

2.7.12 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

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-seção inteira do arquivo Debian é um desperdício de espaço de disco e largura de banda de rede, a implantação de 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.gz`. 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
approx	V:0, I:0	6317	servidor proxy de cache para ficheiros de arquivo Debian (programa OCaml compilado)
apt-cacher	V:0, I:0	289	Proxy de cache para pacotes Debian e ficheiros de código-fonte (programa Perl)
apt-cacher-ng	V:5, I:5	1562	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-key(8)`, `sources.list(5)`, `apt.conf` and `apt_preferences(5)`;
 - `"/usr/share/doc/apt-doc/guide.html/index.html"` e `"/usr/share/doc/apt-doc/offline.html/index.html"` 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.

Here is a rough overview of the key points of the Debian system initialization. Since the Debian system is a moving target, you should refer to the latest documentation.

- [Debian Linux Kernel Handbook](#) is the primary source of information on the Debian kernel.
- `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 por omissão.

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 Stage 1: the UEFI

The [Unified Extensible Firmware Interface \(UEFI\)](#) defines a boot manager as part of the UEFI specification. When a computer is powered on, the boot manager is the 1st stage of the boot process which checks the boot configuration and based on its settings, then executes the specified OS boot loader or operating system kernel (usually boot loader). The boot configuration is defined by variables stored in NVRAM, including variables that indicate the file system paths to OS loaders or OS kernels. An [EFI system partition \(ESP\)](#) is a data storage device partition that is used in computers adhering to the UEFI specification. Accessed by the UEFI firmware when a computer is powered up, it stores UEFI applications and the files these applications need to run, including operating system boot loaders. (On the legacy PC system, [BIOS](#) stored in the [MBR](#) may be used instead.)

3.1.2 Estágio 2: o gestor de arranque

The [boot loader](#) is the 2nd stage of the boot process which is started by the UEFI. It loads the system kernel image and the [initrd](#) image to the memory and hands control over to them. This initrd image is the root filesystem image and its support depends on the bootloader used.

The Debian system normally uses the Linux kernel as the default system kernel. The initrd image for the current 5.x Linux kernel is technically the [initramfs](#) (initial RAM filesystem) image.

There are many boot loaders and configuration options available.

pacote	popcon	tamanho	hoinitrd	gestor de arranque	descrição
grub-efi-amd64	I:214	158	Suportado	GRUB UEFI	This is smart enough to understand disk partitions and filesystems such as vfat, ext4, ... (UEFI)
grub-pc	V:26, I:757	533	Suportado	GRUB 2	This is smart enough to understand disk partitions and filesystems such as vfat, ext4, ... (BIOS)
grub-rescue-pc	V:0, I:1	6380	Suportado	GRUB 2	Isto são imagens de arranque de recuperação do GRUB 2 (CD ou disquete) (Versão PC/BIOS)
lilo	V:0, I:2	697	Suportado	Lilo	Isto baseia-se nas localizações de sectores de dados no disco rígido. (Antigo)
syslinux	V:3, I:46	344	Suportado	Isolinux	Isto compreende o sistema de ficheiros ISO9660. É utilizado pelo CD de arranque.
syslinux	V:3, I:46	344	Suportado	Syslinux	Isto compreende o sistema de ficheiros MSDOS (FAT) . É utilizado pela disquete de arranque.
loadlin	V:0, I:1	90	Suportado	Loadlin	Novo sistema é iniciado a partir do sistema FreeDOS/MSDOS.
mbr	V:0, I:6	50	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



Atenção

Não brinque com os gestores de arranque sem ter discos de arranque de recuperação (caneta USB, CD ou disquete) criados a partir de imagens do pacote `grub-rescue-pc`. Torna-o capaz de arrancar o seu sistema mesmo sem um gestor de arranque funcional no disco rígido.

For GRUB 2, the menu configuration file is located at `"/boot/grub/grub.cfg"` and its key part of menu entry looks like:

```
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
}
```

For this part of `/boot/grub/grub.cfg`, this menu entry means the following.

setting	valor
GRUB2 modules loaded	gzio, part_gpt, ext2
root file system partition used	partition identified by UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1
kernel image path in the root file system	/boot/vmlinuz-5.10.0-6-amd64
kernel boot parameter used	"root=UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1 ro quiet"
initrd image path in the root file system	/boot/initrd.img-5.10.0-6-amd64

Tabela 3.2: The meaning of the menu entry of the above part of `/boot/grub/grub.cfg`

Dica

You can customize GRUB splash image by setting `GRUB_BACKGROUND` variable in `/etc/default/grub` pointing to the image file or placing the image file itself in `/boot/grub/`.

Veja `"info grub"` e `grub-install(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 inspeção 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 disco rígido real.

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 `"/sbin/init"` mas pode ser alterado pelo parâmetro de arranque do kernel como `"init=/path/to/init_program"`.

`"/sbin/init"` is symlinked to `"/lib/systemd/systemd"` after Debian 8 Jessie (released in 2015).

Dica

O comando de iniciação atual do seu sistema pode ser verificado pelo comando `"ps --pid 1 -f"`.

pacote	popcon	tamanho	descrição
systemd	V:826, I:928	16015	daemon de eventos baseado em <code>init(8)</code> para a concorrência (alternativa a <code>sysvinit</code>)
systemd-sysv	V:818, I:925	139	os manuais e ligações necessárias pelo <code>systemd</code> para substituir o <code>sysvinit</code>
init-system-helpers	V:687, I:940	131	ferramentas de ajuda para mudar entre <code>sysvinit</code> e <code>systemd</code>
initscripts	V:77, I:283	176	scripts para inicializar e desligar o sistema
sysvinit-core	V:7, I:9	279	utilitários de <code>init(8)</code> estilo System-V
sysv-rc	V:154, I:296	81	mecanismo de mudança de runlevel estilo System-V
sysvinit-utils	V:459, I:999	79	utilitários estilo System-V (<code>startpar(8)</code> , <code>bootlogd(8)</code> , ...)
lsb-base	V:890, I:999	49	Linux Standard Base funcionalidade de script de <code>init 3.2</code>
insserv	V:204, I:292	150	ferramenta para organizar a sequência de arranque a usar dependências dos scripts <code>init.d</code> LSB
uswsusp	V:3, I:7	714	ferramentas disponibilizadas pelo Linux para utilizar a suspensão de software no espaço de utilizador
kexec-tools	V:1, I:9	288	ferramenta <code>kexec</code> para re-arranques <code>kexec(8)</code> (re-arranque a quente)
systemd-bootchart	V:0, I:1	128	analisador de performance do processo de arranque
bootchart2	V:0, I:0	94	analisador de performance do processo de arranque
pybootchartgui	V:0, I:0	177	analisador de performance do processo de arranque (visualização)
mingetty	V:0, I:3	38	<code>getty(8)</code> apenas de consola
mgetty	V:0, I:1	315	substituto inteligente de modem <code>getty(8)</code>

Tabela 3.3: Lista de utilitários de arranque para o sistema Debian

Dica

Veja [Debian wiki: BootProcessSpeedup](#) para as dicas mais recentes em como acelerar o processo de arranque.

3.2 init do Systemd

Esta secção descreve como o sistema é arrancado pelo programa `systemd(1)` com `PID=1` (i.e., processo `init`).

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.

The spawned processes are placed in individual [Linux control groups](#) named after the unit which they belong to in the private systemd hierarchy (see [cgroups](#) and Secção 4.7.4).

The unit configuration files are loaded from a set of paths (see `systemd-system.conf(5)`) as follows:

- `/lib/systemd/system`: Ficheiros de configuração predefinidos do Sistema Operativo
- `/etc/systemd/system`: ficheiros de configuração do administrador do sistema que se sobrepõem aos ficheiros de configuração predefinidos do Sistema Operativo
- `/run/systemd/system`: ficheiros de configuração gerados em run-time que se sobrepõem aos ficheiros de configuração instalados

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 aparelho exposto em `sysfs(5)` como uma árvore de aparelhos 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 aparelho 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 ligação 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/rc[0123]` 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".

3.2.1 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.

3.2.2 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.

3.2.3 Inicialização da interface de rede

As interfaces de rede são tipicamente iniciadas em `"networking.service"` para a interface `lo` e `"NetworkManager.service"` para as outras interfaces em sistemas Debian modernos sob `systemd`.

Veja Capítulo 5 para saber como os configurar.

3.3 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 significativa
6	KERN_INFO	informativa
7	KERN_DEBUG	mensagens de nível de depuração

Tabela 3.4: Lista de níveis de erro do kernel

3.4 A mensagem do sistema

Under `systemd`, both kernel and system messages are logged by the journal service `systemd-journald.service` (a.k.a `journald`) either into a persistent binary data below `"/var/log/journal"` or into a volatile binary data below `"/run/log/journal"`. These binary log data are accessed by the `journalctl(1)` command. For example, you can display log from the last boot as:

Operação	Fragmentos de comando
View log for system services and kernel from the last boot	<code>"journalctl -b --system"</code>
View log for services of the current user from the last boot	<code>"journalctl -b --user"</code>
View job log of "\$unit" from the last boot	<code>"journalctl -b -u \$unit"</code>
View job log of "\$unit" ("tail -f" style) from the last boot	<code>"journalctl -b -u \$unit -f"</code>

Tabela 3.5: List of typical `journalctl` command snippets

```
$ journalctl -b
```

Under `systemd`, the system logging utility `rsyslogd(8)` may be uninstalled. If it is installed, it changes its behavior to read the volatile binary log data (instead of pre-`systemd` default `"/dev/log"`) and to create traditional permanent ASCII system log data. This can be customized by `"/etc/default/rsyslog"` and `"/etc/rsyslog.conf"` for both the log file and on-screen display. See `rsyslogd(8)` and `rsyslog.conf(5)`. See also Secção 9.3.2.

3.5 System management

The `systemd` offers not only init system but also generic system management operations with the `systemctl(1)` command.

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}|$aparelho"` 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".

3.6 Other system monitors

Here are a list of other monitoring command snippets under `systemd`. Please read the pertinent manpages including `cgroups(7)`.

3.7 Personalizar o systemd

3.7.1 Socket activation

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.

Operação	Fragmentos de comando
Lista toda a configuração da unidade alvo	"systemctl list-units --type=target"
Lista toda a configuração da unidade de serviço	"systemctl list-units --type=service"
Lista todos os tipos de configuração da unidade	"systemctl list-units --type=help"
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 aparelho"	"systemctl status \$unit \$PID \$device"
Mostra propriedades de "\$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 variável=valor"
Remove a definição da "variável" de ambiente de função	"systemctl unset-environment variável"
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: List of typical systemctl command snippets

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"
Show terse runtime status information of the user of the caller's session	"loginctl user-status"
Show terse runtime status information of the caller's session	"loginctl session-status"
Track boot process by the cgroups	"systemd-cgls"
Track boot process by the cgroups	"ps xawf -eo pid,user,cgroup,args"
Track boot process by the cgroups	Lê sysfs sob "/sys/fs/cgroup/systemd/"

Tabela 3.7: List of other monitoring command snippets under systemd

```
$ sudo systemctl stop sshd.service
$ sudo systemctl mask sshd.service
```

The on-demand socket activation system of the classic Unix services was through the `inetd` (or `xinetd`) superserver. Under `systemd`, the equivalent can be enabled by adding `*.socket` and `*.service` unit configuration files.

`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
```

3.8 O sistema udev

The [udev system](#) provides mechanism for the automatic hardware discovery and initialization (see [udev\(7\)](#)) since Linux kernel 2.6. Upon discovery of each device by the kernel, the udev system starts a user process which uses information from the [sysfs](#) filesystem (see Secção 1.2.12), loads required kernel modules supporting it using the [modprobe\(8\)](#) program (see Secção 3.8.1), and creates corresponding device nodes.

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 aparelhos não precisam de ser os estáticos. Pode usar o [UUID](#) para montar os aparelhos em vez dos nomes de aparelho 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.

3.8.1 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 [3.8](#)) 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 aparelhos de rede Point-to-Point virtuais (TUN) e aparelhos de rede Ethernet virtuais (TAP),
- módulos [netfilter](#) que disponibilizam capacidades de firewall `netfilter` (`iptables(8)`, Secção [5.6](#)) 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 aparelho especial ao recompilar o kernel. Veja Secção [9.10](#).

Capítulo 4

Authentication and access controls

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
<code>/etc/passwd</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	informação da conta do utilizador (filtrada)
<code>/etc/shadow</code>	<code>-rw-r-----</code>	<code>root</code>	<code>shadow</code>	informação segura da conta do utilizador
<code>/etc/group</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	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 <code>/etc/shadow</code>
	nenhum login para esta conta
!	nenhum login para esta conta

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:$1$VGZLVbS$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 expirado 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 <i>nome_de_utilizador</i>	navegar na informação da conta de " <i>nome_de_utilizador</i> "
getent shadow <i>nome_de_utilizador</i>	explorar informação confidencial (shadow) da conta de " <i>nome_de_utilizador</i> "
getent group <i>nome_de_grupo</i>	navegar na informação do grupo de " <i>nome_do_grupo</i> "
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:32, I:371	364	mkpasswd	frontend cheio de funcionalidades para a biblioteca <code>crypt(3)</code>
openssl	V:814, I:994	1465	<code>openssl passwd</code>	computa hashes de palavras-passe (OpenSSL). <code>passwd(1ssl)</code>

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 envolve 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:813, I:999	1048	Pluggable Authentication Modules (serviço básico)
libpam-ldap	I:12	249	Pluggable Authentication Module que permite interfaces LDAP
libpam-cracklib	I:15	117	Pluggable Authentication Module para ativar suporte a cracklib
libpam-systemd	V:484, I:869	574	Pluggable Authentication Module para registar sessões de utilizador para o <code>logind</code>
libpam-doc	I:1	1046	Pluggable Authentication Modules (documentação em html e texto)
libc6	V:946, I:999	12772	GNU C Library: Bibliotecas de partilha que também disponibilizam o serviço "Name Service Switch"
glibc-doc	I:11	3161	Biblioteca GNU C: Manuais (Manpages)
glibc-doc-reference	I:4	12740	Biblioteca GNU C: Manual de referência em info, pdf e html (não-livre)
libnss-mdns	I:508	150	Módulo NSS para resolução de nomes Multicast DNS
libnss-ldap	I:11	265	Módulo NSS para utilizar o LDAP como um serviço nomeador
libnss-ldapd	I:14	153	Módulo NSS para usar o LDAP como um serviço nomeador (nova bifurcação de <code>libnss-ldap</code>)

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
<code>/etc/pam.d/nome_do_programa</code>	defina a configuração do PAM para o programa <code>"nome_do_programa"</code> ; veja <code>pam(7)</code> e <code>pam.d(5)</code>
<code>/etc/nsswitch.conf</code>	define a configuração NSS com a entrada para cada serviço. Veja <code>nsswitch.conf(5)</code>
<code>/etc/nologin</code>	limita o login de utilizador pelo módulo <code>pam_nologin(8)</code>
<code>/etc/securetty</code>	limita a tty ao acesso de root pelo módulo <code>pam_securetty(8)</code>
<code>/etc/security/access.conf</code>	define limites de acesso pelo módulo <code>pam_access(8)</code>
<code>/etc/security/group.conf</code>	define a restrição baseada em grupo pelo módulo <code>pam_group(8)</code>
<code>/etc/security/pam_env.conf</code>	define as variáveis de ambiente pelo módulo <code>pam_env(8)</code>
<code>/etc/environment</code>	define variáveis de ambiente adicionais pelo módulo <code>pam_env(8)</code> com o argumento <code>"readenv=1"</code>
<code>/etc/default/locale</code>	define o locale pelo módulo <code>pam_env(8)</code> com o argumento <code>"readenv=1 envfile=/etc/default/locale"</code> (Debian)
<code>/etc/security/limits.conf</code>	define restrição de recursos (<code>ulimit</code> , <code>core</code> , ...) pelo módulo <code>pam_limits(8)</code>
<code>/etc/security/time.conf</code>	define a retenção de tempo pelo módulo <code>pam_time(8)</code>
<code>/etc/systemd/logind.conf</code>	define a configuração do gestor de login do <code>systemd</code> (veja <code>logind.conf(5)</code> e <code>systemd-logind.service(8)</code>)

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 directory Access Protocol \(LDAP\)](#) para administrar muitos sistemas tipo-Unix e não-tipo-Unix na rede. A implementação de código aberto do Lightweight directory 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-ldap` e `libnss-ldap`. 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"`.
 - Debian utiliza `"/etc/pam_ldap.conf"` como ficheiro de configuração para `libpam-ldap` e `"/etc/pam_ldap.secret"` como ficheiro para armazenar a palavra-passe do root.
- Altera a configuração do NSS no ficheiro `"/etc/nsswitch.conf"` para usar `"ldap"` em vez da predefinição (`"compat"` ou `"file"`).
 - Debian utiliza o `"/etc/libnss-ldap.conf"` como o ficheiro de configuração para `libnss-ldap`.
- Tem de fazer `libpam-ldap` para usar a ligação [SSL \(ou TLS\)](#) para a segurança da palavra-passe.
- Pode fazer a `libnss-ldap` 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 `pam_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 recadeiar 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

Installing the `libpam-cracklib` package enables you to force stricter password rule.

On a typical GNOME system which automatically installs `libpam-gnome-keyring`, `"/etc/pam.d/common-password"` looks like:

```
# 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 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 amigável 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.)

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 políticas de firewall (veja Secção [5.6](#)) juntamente com as seguintes ferramentas de segurança podem melhorar a situação de segurança.

pacote	popcon	tamanho	descrição
knockd	V:0, I:3	102	pequeno daemon port-knock knockd(1) e cliente konck(1)
fail2ban	V:116, I:129	2092	banir IPs que causam vários erros de autenticação
libpam-shield	V:0, I:0	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 acessem à sua máquina com privilégios de root, precisa de tomar as seguintes acções.

- Prevenir acesso físico ao disco rígido
- Lock UEFI/BIOS and prevent booting from the removable media
- Definir palavra-passe para sessão interactiva do GRUB
- Bloquear o menu do GRUB de ser editado

Com acesso físico ao disco rígido, redefinir a palavra-passe é relativamente fácil com os seguintes passos.

1. Move the hard disk to a PC with CD bootable UEFI/BIOS.
2. Arrancar o sistema com um suporte de recuperação (disco de arranque Debian, CD Knoppix, CD GRUB, ...).
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/hda6 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 9.5.11). 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

There are access controls to the system other than the password based authentication and file permissions.

Nota

Veja Secção 9.4.15 para recadeiair a funcionalidade do kernel [chave de atenção segura \(SAK\)](#).

4.7.1 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"`.

4.7.2 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)`.

4.7.3 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`
- Systemd service unit configuration for [daemon](#)
- [PAM \(Pluggable Authentication Modules\)](#)
- `/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](#)

See Secção 3.5, Secção 4.5.1, and Secção 5.6.

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.)

4.7.4 Linux security features

Linux kernel has evolved and supports security features not found in traditional UNIX implementations.

Linux supports [extended attributes](#) which extend the traditional UNIX attributes (see `xattr(7)`).

Linux divides the privileges traditionally associated with superuser into distinct units, known as [capabilities\(7\)](#), which can be independently enabled and disabled. Capabilities are a per-thread attribute since kernel version 2.2.

The [Linux Security Module \(LSM\) framework](#) provides a [mechanism for various security checks](#) to be hooked by new kernel extensions. For example:

- [AppArmor](#)
- [Security-Enhanced Linux \(SELinux\)](#)
- [Smack \(Simplified Mandatory Access Control Kernel\)](#)
- [Tomoyo Linux](#)

Since these extensions may tighten privilege model tighter than the ordinary Unix-like security model policies, even the root power may be restricted. You are advised to read the [Linux Security Module \(LSM\) framework document at kernel.org](#).

Linux [namespaces](#) wrap a global system resource in an abstraction that makes it appear to the processes within the namespace that they have their own isolated instance of the global resource. Changes to the global resource are visible to other processes that are members of the namespace, but are invisible to other processes. Since kernel version 5.6, there are 8 kinds of namespaces (see `namespaces(7)`, `unshare(1)`, `nsenter(1)`).

As of Debian 11 Bullseye (2021), Debian uses unified cgroup hierarchy (a.k.a. [cgroups-v2](#)).

Usage examples of [namespaces](#) with [cgroups](#) to isolate their processes and to allow resource control are:

- [Systemd](#). See Secção 3.2.
- [Sandbox environment](#). See Secção 7.6.
- [Linux containers](#) such as [Docker](#), [LXC](#). See Secção 9.11.

These functionalities can't be realized by Secção 4.1. These advanced topics are mostly out-of-scope for this introductory document.

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.

Por exemplo, `/etc/hosts` parece-se com o seguinte.

```
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:360, I:426	15210	config::NM	NetworkManager (daemon): gere a rede automaticamente
network-manager-gnome	V:124, I:357	5351	config::NM	NetworkManager (frontend do GNOME)
ifupdown	V:612, I:989	217	config::ifupdown	ferramenta standard para ativar e desativar a rede (especifico de Debian)
isc-dhcp-client	V:222, I:980	686	config::low-level	Cliente DHCP
pppoeconf	V:0, I:8	192	config::helper	ajudante de configuração para ligação PPPoE
wpasupplicant	V:335, I:491	3439	, ,	suporte de cliente para WPA e WPA2 (IEEE 802.11i)
wpaui	V:0, I:2	780	, ,	Cliente GUI Qt para o wpa_supplicant
wireless-tools	V:180, I:239	297	, ,	ferramentas para manipular Extensões Wireless do Linux
iw	V:320, I:467	293	, ,	tool for configuring Linux wireless devices
iproute2	V:695, I:937	2867	config::iproute2	iproute2 , IPv6 e outras configurações avançadas de rede: ip(8) , tc(8) , etc
iptables	V:314, I:990	2521	config::Netfilter	ferramentas administrativas para filtragem de pacotes e NAT (Netfilter)
iputils-ping	V:224, I:997	113	test	teste de acessibilidade de rede de uma máquina remota pelo nome-de-máquina ou endereço IP (iproute2)
iputils-arping	V:7, I:98	55	test	teste de acessibilidade de rede de uma máquina remota especificado pelo endereço ARP
iputils-tracpath	V:4, I:53	72	test	rastreia o caminho de rede até uma máquina remota
ethtool	V:103, I:266	597	test	mostra ou altera as definições de um aparelho Ethernet
mtr-tiny	V:6, I:55	161	test::low-level	rastreia o caminho de rede até uma máquina remota (curses)
mtr	V:4, I:46	214	, ,	rastreia o caminho de rede até uma máquina remota (curses e GTK)
gnome-nettool	V:1, I:38	2105	, ,	ferramentas para operações comuns de informação de rede (GNOME)
nmap	V:27, I:246	4509	, ,	mapeamento de rede / sondagem de portos (Nmap , consola)
zenmap	V:1, I:8	2939	, ,	mapeamento de rede / sondagem de portos (GTK)
tcpdump	V:19, I:208	1329	, ,	analizador de tráfego de rede (Tcpdump , consola)
wireshark	I:54	65	, ,	analizador de tráfego de rede (Wireshark , GTK)
tshark	V:3, I:33	408	, ,	analizador de tráfego de rede (consola)
tcptrace	V:0, I:3	401	, ,	produz um sumário das ligações a partir da saída do tcpdump
snort	V:0, I:1	2206	, ,	sistema flexível de detecção de intrusos na rede (Snort)
ntopng	V:1, I:2	969	, ,	mostra a utilização da rede num navegador web
dnsutils	V:59, I:490	261	, ,	clientes de rede disponibilizados com BIND : nslookup(8) , nsupdate(8) , dig(8)
dlint	V:0, I:6	53	, ,	verifica informação de zona DNS a usar pesquisas do servidor de nomes
dnstracer	V:0, I:1	61	, ,	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` coincide com o nome da máquina definido em `"/etc/hostname"`.

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 o PC estação de trabalho do ambiente LAN adhoc típico, o nome de máquina pode ser resolvido via Multicast DNS (mDNS, [Zeroconf](#)) adicionalmente aos métodos `file` e `dns` básicos.

- [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=return] dns mdns4"`.
- Os nomes de máquinas com o [domínio de pseudo nível de topo](#) (TLD) `".local"` estão resolvidos.
- O endereço multicast link-local mDNS IPv4 `"224.0.0.251"` ou o equivalente dele em IPv6 `"FF02::FB"` são usados para fazer pesquisas DNS para nomes que terminem com `".local"`.

A resolução de nome de máquina através do depreciado [NETBios sobre TCP/IP](#) utilizada em antigos sistemas Windows pode ser disponibilizada ao instalar o pacote `winbind`. O ficheiro `"/etc/nsswitch.conf"` deverá ter um bloco como `"hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4 wins"` para ativar esta funcionalidade. (Os sistemas modernos de Windows normalmente utilizam o método `dns` para a resolução dos nomes de máquinas.)

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.

5.1.2 O nome da interface de rede

The `systemd` uses ["Predictable Network Interface Names"](#) such as `"enp0s25"`.

Classe	endereços de rede	máscara de rede	máscara de rede /bits	of subnets
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

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

IP address written with colon are [IPv6 address](#), e.g., "::<1" for localhost.

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 aparelhos de rede

Apesar da maioria dos aparelhos de hardware serem suportados pelo sistema Debian, existem alguns aparelhos 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.service" para as outras interfaces em sistemas Debian modernos sob systemd.

Debian can manage the network connection via management [daemon](#) software such as [NetworkManager \(NM\)](#) (network-manager and associated packages).

- 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

Official documentations for NM on Debian are provided in `"/usr/share/doc/network-manager/README.Debian"`.

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 adduser foo netdev
```

2. Mantenha a configuração de `"/etc/network/interfaces"` tão simples como o seguinte.

```
auto lo
iface lo inet loopback
```

3. Restart NM by the following.

```
$ sudo systemctl restart network-manager
```

4. Configure a sua rede através da GUI.

Nota

Only interfaces which are **not** listed in `"/etc/network/interfaces"` are managed by NM to avoid conflict with `ifupdown`.

Dica

If you wish to extend network configuration capabilities of NM, please seek appropriate plug-in modules and supplemental packages such as `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 de rede de baixo nível

For the low level network configuration on Linux, use the [iproute2](#) programs (`ip(8)`, ...).

5.4.1 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
<code>ifconfig(8)</code>	<code>ip addr</code>	endereço de protocolo (IP ou IPv6) num aparelho
<code>route(8)</code>	<code>ip route</code>	entrada na tabela de rotas
<code>arp(8)</code>	<code>ip neigh</code>	entrada na cache ARP ou NDISC
<code>ipmaddr</code>	<code>ip maddr</code>	endereço multicast
<code>iptunnel</code>	<code>ip tunnel</code>	túnel sobre IP
<code>nameif(8)</code>	<code>ifrename(8)</code>	nomeia as interfaces de rede baseadas no endereço MAC
<code>mii-tool(8)</code>	<code>ethtool(8)</code>	Definições de aparelho Ethernet

Tabela 5.3: Tabela de tradução dos comandos obsoletos `net-tools` para os novos comandos `iproute2`

Vea `ip(8)` e [Manual do Conjunto de Utilitários IPROUTE2](#).

5.4.2 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
<code>ip addr show</code>	mostra o estado de ligação e endereço das interfaces ativas
<code>route -n</code>	mostra toda a tabela de rotas em endereços numéricos
<code>ip route show</code>	mostra toda a tabela de rotas em endereços numéricos
<code>arp</code>	mostra o conteúdo atual das tabelas de cache ARP
<code>ip neigh</code>	mostra o conteúdo atual das tabelas de cache ARP
<code>plog</code>	mostra o log do daemon ppp
<code>ping yahoo.com</code>	verifica a ligação de Internet para "yahoo.com"
<code>whois yahoo.com</code>	verifica quem registou "yahoo.com" na base de dados de domínios
<code>traceroute yahoo.com</code>	rastreia a ligação Internet até "yahoo.com"
<code>tracepath yahoo.com</code>	rastreia a ligação Internet até "yahoo.com"
<code>mtr yahoo.com</code>	rastreia a ligação Internet até "yahoo.com" (repetidamente)
<code>dig [@servidor-dns.com] exemplo.com [{a mx any}]</code>	verifica os registos DNS de "exemplo.com" pelo "servidor-dns.com" para um registo "a", "mx", ou "any"
<code>iptables -L -n</code>	verifica o filtro de pacotes
<code>netstat -a</code>	procura todos os portos abertos
<code>netstat -l --inet</code>	procura portos a escutar
<code>netstat -ln --tcp</code>	procura portos TCP a escutar (numérico)
<code>dlint exemplo.com</code>	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 `/sbin/`. Pode necessitar de escrever o caminho do comando completo tal como `/sbin/ifconfig` ou adicionar `/sbin` à lista `$PATH` no seu `~/ .bashrc`.

5.5 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:8, I:118	97	mostra informação da utilização de largura de banda numa interface de rede
iperf	V:4, I:55	305	ferramenta de medição da largura de banda do Protocolo Internet
ifstat	V:0, I:8	60	InterFace STATistics Monitoring
bmon	V:1, I:18	146	monitor de largura de banda portátil e estimador de taxas
ethstatus	V:0, I:4	40	script que mede rapidamente a transferência efectiva de um aparelho de rede
bing	V:0, I:1	80	testador de largura de banda empírica estocástica
bwm-ng	V:2, I:18	90	monitor de largura de banda pequeno e simples baseado em consola
ethstats	V:0, I:0	23	monitor de estatísticas de Ethernet baseado em consola
ipfm	V:0, I:0	78	ferramenta de análise de largura de banda

Tabela 5.5: Lista de ferramentas de optimização de rede

5.5.1 Encontrar o MTU óptimo

O valor [Maximum Transmission Unit \(MTU\)](#) pode ser determinado experimentalmente com `ping(8)` com a opção `-M do` a qual envia pacotes ICMP com tamanho de dados inicial de 1500 (com offset de 28 bytes para o cabeçalho IP+ICMP) e encontra o tamanho maior sem fragmentação IP.

Por exemplo, tente o seguinte:

```
$ ping -c 1 -s $((1500-28)) -M do www.debian.org
PING www.debian.org (194.109.137.218) 1472(1500) bytes of data.
From 192.168.11.2 icmp_seq=1 Frag needed and DF set (mtu = 1454)

--- www.debian.org ping statistics ---
0 packets transmitted, 0 received, +1 errors
```

Try MTU=1454 instead of MTU=1500

You see `ping(8)` succeed with MTU=1454.

If MTU is not 1500, you may want to configure MTU settings in NM.

Este processo é a descoberta do [Caminho MTU \(PMTU\) \(RFC1191\)](#) e o comando `tracert(8)` pode automatizar isto.

Dica

O exemplo acima com valor PMTU de 1454 é para o meu provedor de FTTP anterior o qual utilizava [Modo de Transferência Assíncrona](#) (ATM) com a coluna vertebral da rede dele e servia os seus clientes com [PPPoE](#). O valor de PMTU real depende do seu ambiente, p.e. 1500 para o meu provedor de FTTP novo.

ambiente de rede	MTU	racional
Ligação Dial-up (IP: PPP)	576	standard
Ligação Ethernet (IP: DHCP ou fixo)	1500	standard e predefinido
Ligação Ethernet (IP: PPPoE)	1492 (=1500-8)	2 bytes para o cabeçalho de PPP e 6 bytes para o cabeçalho de PPPoE
Ligação Ethernet (espinha dorsal do ISP: ATM, IP: DHCP ou fixo)	1462 (=48*31-18-8)	especulação do autor: 18 bytes para cabeçalho de Ethernet, 8 bytes para trailer de SAR
Ligação Ethernet (espinha dorsal do ISP: ATM, IP: PPPoE)	1454 (=48*31-8-18-8)	author's speculation: 18 bytes for Ethernet header, 8 bytes for SAR trailer, 2 bytes for PPP header and 6 bytes for PPPoE header

Tabela 5.6: Regras básicas para o valor MTU ótimo

Adicionalmente a estas regras básicas, deve saber o seguinte:

- Qualquer utilização de métodos de túnel ([VPN](#) etc.) pode reduzir o MTU ótimo 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 otimização baseada no `iptables(8)` (veja [Secção 5.6](#)) pode apertar o tamanho do pacote pelo MSS e é útil para o router. Veja "TCP MSS" em `iptables(8)`.

5.5.2 Otimização WAN TCP

Para a WAN moderna de alta largura de banda e alta latência, a performance do TCP pode ser maximizada a ajustar os parâmetros de tamanho de buffer TCP conforme descrito em "[Guia de Ajustes de TCP](#)" e em "[Ajustes de TCP](#)". Até agora, os ajustes predefinidos atuais de Debian servem bem mesmo para a minha LAN ligada pelo serviço FTTP rápido de 1G bps.

5.6 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 3.8.1](#)).

O principal programa de utilizador para o [netfilter](#) é o `iptables(8)`. Pode configurar manualmente e interativamente o [netfilter](#) a partir da shell, gravar o estado dele com `iptables-save(8)` e restaurá-lo via script init com `iptables-restore(8)` após o reiniciar do sistema.

Scripts de ajuda de configuração como o [shorewall](#) facilitam este processo.

Veja documentação em <http://www.netfilter.org/documentation/> (ou em `/usr/share/doc/iptables/html/`).

- [Manual de Conceitos de Rede em Linux](#)

pacotes	popcon	tamanho	descrição
iptables	V:314, I:990	2521	ferramentas de administração para netfilter (iptables(8) para IPv4, ip6tables(8) para IPv6)
arptables	V:0, I:2	96	ferramentas de administração para netfilter (arptables(8) para ARP)
ebtables	V:16, I:37	265	ferramentas de administração para netfilter (ebtables(8) para criação de pontes Ethernet)
iptstate	V:0, I:3	116	monitoriza continuamente o estado do netfilter (semelhante ao top(1))
shorewall-init	V:0, I:0	68	Inicialização de Shoreline Firewall
shorewall	V:4, I:12	2458	Shoreline Firewall , gerador de ficheiro de configuração netfilter
shorewall-lite	V:0, I:0	65	Shoreline Firewall , gerador de ficheiro de configuração (versão leve) netfilter
shorewall6	V:1, I:2	779	Shoreline Firewall , gerador de ficheiro de configuração (versão IPv6) netfilter
shorewall6-lite	V:0, I:0	64	Shoreline Firewall , gerador de ficheiro de configuração (IPv6, versão light) netfilter

Tabela 5.7: Lista de ferramentas de firewall

- [Manual do Packet Filtering em Linux 2.4](#)
- [Como Fazer NAT em Linux 2.4](#)

Dica

Apesar destes terem sido escritos para o Linux **2.4**, ambos comandos [iptables\(8\)](#) e função de kernel [netfilter](#) aplicam-se às séries **2.6** e **3.x** do kernel Linux.

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:54, I:134	192549	X	Chromium , (browser open-source da Google)
firefox	V:13, I:20	209827	, ,	Firefox , (navegador de código aberto da Mozilla, apenas disponível no Debian Unstable)
firefox-esr	V:199, I:421	200011	, ,	Firefox ESR , (Lançamento de Suporte Estendido do Firefox)
epiphany-browser	V:4, I:23	3730	, ,	GNOME , HIG compliant, Epiphany
konqueror	V:17, I:95	20763	, ,	KDE , Konqueror
dillo	V:1, I:7	1536	, ,	Dillo , (navegador leve, baseado em FLTK)
w3m	V:25, I:252	2367	texto	w3m
lynx	V:11, I:97	1948	, ,	Lynx
elinks	V:5, I:27	1767	, ,	ELinks
links	V:5, I:38	2250	, ,	Ligações (apenas texto)
links2	V:1, I:14	5418	gráficos	Ligações (gráficos de consola sem X)

Tabela 6.1: Lista de exploradores web

6.1.1 Configuração do explorador

Poderá ser capaz de utilizar as seguintes cadeias especiais de URL para alguns navegadores para confirmar as suas definições.

- "about:"
- "about:config"
- "about:plugins"

Debian oferece muitos pacotes livres com plugins de navegador na área de arquivo main, os quais podem lidar não apenas com [Java \(plataforma de software\)](#) e [Flash](#) mas também com ficheiros [MPEG](#), [MPEG2](#), [MPEG4](#), [DivX](#), [Windows Media Video \(.wmv\)](#), [QuickTime \(.mov\)](#), [MP3 \(.mp3\)](#), [Ogg/Vorbis](#), DVDs, VCDs, etc. Debian também oferece programas de ajuda para instalar pacotes não-livres de plugins de navegador nas áreas de arquivo contrib ou non-free.

pacote	popcon	tamanho	área	descrição
pepperflashplugin-nonfree	V:0, I:15	26	contrib	Pepper Flash Player - complemento de navegadores

Tabela 6.2: Lista de pacotes de plugins de navegadores

Dica

Apesar do uso dos pacotes Debian em cima ser mais fácil, os plugins do navegador podem ainda ser adicionados manualmente ao instalar os "*.so" nos diretórios de plugins (ex. "/usr/lib/iceweasel/plugins/") e a reiniciar os navegadores.

Alguns sites web recusam a ligação conforme a cadeia user-agent do seu navegador. Pode contornar esta situação ao [enganar a cadeia user-agent](#). Por exemplo, pode fazê-lo ao adicionar a seguinte linha em ficheiros de configuração do utilizador como o "~/.gnome2/epiphany/mozilla/epiphany/user.js" ou "~/.mozilla/firefox/*.default/user.js".

```
user_pref("general.useragent.override", "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0) ↵";
```

Alternativamente, pode adicionar ou reiniciar esta variável ao escrever "about:config" no URL e clicar com o botão direito no conteúdo mostrado.



Cuidado

Uma cadeia user-agent enganada pode causar [maus efeitos colaterais com Java](#).

6.2 O sistema de correio electrónico (mail)

This section focuses on typical mobile workstations on consumer grade Internet connections.



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.)
- The email message format covering header and body data is extended by [Multipurpose Internet Mail Extensions \(MIME\)](#) from the plain ASCII text to other character encodings, as well as attachments of audio, video, images, and application programs.

Full featured GUI based [email clients](#) offer all the following functions using the GUI based intuitive configuration.

- It creates and interprets the message header and body data using [Multipurpose Internet Mail Extensions \(MIME\)](#) to deal the content data type and encoding.
- It authenticates itself to the ISP's SMTP and IMAP servers using the legacy [basic access authentication](#) or modern [OAuth 2.0](#). (For [OAuth 2.0](#), set it via Desktop environment settings. E.g., "Settings" -> "Online Accounts".)
- It sends the message to the ISP's smarthost SMTP server listening to the message submission port (587).
- It receives the stored message on the ISP's server from the TLS/IMAP4 port (993).
- It can filter mails by their attributes.
- It may offer additional functionalities: Contacts, Calendar, Tasks, Memos.

pacote	popcon	tamanho	tipo
evolution	V:29, I:219	475	Programa de GUI X (GNOME3, conjunto groupware)
thunderbird	V:57, I:134	168690	X GUI program (GTK, Mozilla Thunderbird)
kmail	V:33, I:85	18011	Programa de GUI X (KDE)
mutt	V:33, I:278	7071	programa de terminal de caracteres provavelmente usado com o vim
mew	V:0, I:0	2334	programa de terminal de caracteres sob (x) emacs

Tabela 6.3: Lista de agentes utilizador de mail (MUA)

6.2.2 Modern mail service limitation

Modern mail service are under some limitations in order to minimize exposure to the spam (unwanted and unsolicited email) problems.

- It is not realistic to run SMTP server on the consumer grade network to send mail directly to the remote host reliably.
- A mail may be rejected by any host en route to the destination quietly unless it appears as authentic as possible.
- Não é realista para a confiança da máquina remota, esperar que um único smarthost envie mails com um endereço de mail fonte não relacionado.

This is because:

- The SMTP port (25) connections from hosts serviced by the consumer grade network to the Internet are blocked.
- The SMTP port (25) connections to hosts serviced by the consumer grade network from the Internet are blocked.

- The outgoing messages from hosts serviced by the consumer grade network to the Internet can only be sent via the message submission port (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.
- The smarthost may rewrite the source mail address in the message header to your mail account on the smarthost to prevent email address spoofing.

6.2.3 Historic mail service expectation

Some programs on Debian expect to access the `/usr/sbin/sendmail` command to send emails as their default or customized setting since the mail service on a UNIX system functioned historically as:

- An email is created as a text file.
- The email is handed to the `/usr/sbin/sendmail` command.
- For the destination address on the same host, the `/usr/sbin/sendmail` command makes local delivery of the email by appending it to the `/var/mail/$username` file.
 - Commands expecting this feature: `apt-listchanges`, `cron`, `at`, ...
- For the destination address on the remote host, the `/usr/sbin/sendmail` command makes remote transfer of the email to the destination host found by the DNS MX record using SMTP.
 - Commands expecting this feature: `popcon`, `reportbug`, `bts`, ...

6.2.4 Agente de transporte de mail (MTA)

In principle, mobile workstations should function without the `/usr/sbin/sendmail` command provided by the [mail transfer agent \(MTA\)](#) program.

The Debian system usually installs MTA to cope with [Secção 6.2.2](#) and [Secção 6.2.3](#) even if mobile workstations installed full featured GUI based [email clients](#).

For mobile workstations, the typical choice of MTA is either `exim4-daemon-light` or `postfix` with its installation option such as "Mail sent by smarthost; received via SMTP or fetchmail" selected. These are light weight MTAs that respect `/etc/aliases`.

Dica

Configuring `exim4` to send the Internet mail via multiple corresponding smarthosts for multiple source email addresses is non-trivial. If you need such capability for some programs, set them up to use `msmtp` which is easy to set up for multiple source email addresses. Then leave main MTA only for a single email address.

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
```

pacote	popcon	tamanho	descrição
exim4-daemon-light	V:320, I:346	1517	Agente de transporte de mail Exim4 (MTA: predefinido em Debian)
exim4-daemon-heavy	V:7, I:8	1668	Exim4 mail transport agent (MTA: flexible alternative)
exim4-base	V:328, I:355	1720	Documentação do Exim4 (texto) e ficheiros comuns
exim4-doc-html	I:1	3667	Documentação do Exim4 (html)
exim4-doc-info	I:1	625	Documentação do Exim4 (info)
postfix	V:147, I:162	4183	Postfix mail transport agent (MTA: secure alternative)
postfix-doc	I:9	4444	Documentação do Postfix (html+texto)
saslm2-bin	V:5, I:19	428	Implementação Cyrus SASL API (postfix suplementar para SMTP AUTH)
cyrus-saslm2-doc	I:1	575	Cyrus SASL - documentação
msmtp	V:6, I:12	547	Light weight MTA
msmtp-mta	V:4, I:5	86	Light weight MTA (sendmail compatibility extension to msmtp)
esmtm	V:0, I:0	128	Light weight MTA
esmtm-run	V:0, I:0	32	Light weight MTA (sendmail compatibility extension to esmtm)
nullmailer	V:8, I:10	479	Strip down MTA, no local mail
ssmtm	V:6, I:10	2	Strip down MTA, no local mail
sendmail-bin	V:14, I:15	1859	Full featured MTA (only if you are already familiar)
courier-mta	V:0, I:0	2442	Full featured MTA (web interface etc.)

Tabela 6.4: List of basic mail transport agent related packages

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".

Select "No" for "Hide local mail name in outgoing mail?". (Use "/etc/email-addresses" as in Secção 6.2.4.3, instead.)

Responda a "Minimizar a quantidade de consultas DNS (Chamar-a-Pedido)?" como uma das seguintes.

- "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".

Select "Yes" for "Split configuration into small files?:".

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 exim4(8) with "QUEUERUNNER='queueonly'", "QUEUERUNNER='nodaemon'", etc. in "/etc/default/exim4" to minimize system resource usages. (optional)

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.template"` (configuração não-dividida)
 - criar novos ficheiros ou editar ficheiros existentes nos sub-diretórios `"/etc/exim4/exim4.conf.d"`. (configuração dividida)
- Run `"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

For all practical consideration, use **SMTP** with **STARTTLS** on port 587 or **SMTPS** SSL (SMTPS) on port 465, instead of plain SMTP on port 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
<code>postfix(1)</code>	Programa de controlo do postfix
<code>postconf(1)</code>	Utilitário de configuração do postfix
<code>postconf(5)</code>	Parâmetros de configuração do postfix
<code>postmap(1)</code>	Manutenção da tabela de buscas do Postfix
<code>postalias(1)</code>	Manutenção da base de dados de alias do Postfix

Tabela 6.5: 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 de mail [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 6.6: 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 6.7: 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-`J`).

pacote	popcon	tamanho	ferramenta	descrição
openssh-client	V:809, I:997	4298	<code>ssh(1)</code>	Cliente de shell segura
openssh-server	V:701, I:844	1567	<code>sshd(8)</code>	Servidor de shell segura
ssh-askpass	V:2, I:32	106	<code>ssh-askpass(1)</code>	pede ao utilizador uma frase passe para <code>ssh-add</code> (X simples)
ssh-askpass-gnome	V:0, I:5	294	<code>ssh-askpass-gnome(1)</code>	asks user for a pass phrase for <code>ssh-add</code> (GNOME)
ssh-askpass-fullscreen	V:0, I:0	42	<code>ssh-askpass-fullscreen(1)</code>	asks user for a pass phrase for <code>ssh-add</code> (GNOME) with extra eye candy
shellinabox	V:0, I:2	507	<code>shellinaboxd(1)</code>	web server for browser accessible VT100 terminal emulator

Tabela 6.8: Lista de servidores de acesso remoto e utilitários

Although `shellinabox` is not a SSH program, it is listed here as an interesting alternative for the remote terminal access.

See also Secção [7.8](#) for connecting to remote X client programs.



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

The OpenSSH SSH daemon supports SSH protocol 2 only.

Please read `"/usr/share/doc/openssh-client/README.Debian.gz"`, `ssh(1)`, `sshd(8)`, `ssh-agent(1)`, and `ssh-keygen(1)` and `ssh-add(1)` and `ssh-agent(1)`.



Atenção

o `"/etc/ssh/sshd_not_to_be_run"` não pode estar presente se desejar correr o servidor OpenSSH. Don't enable rhost based authentication (`HostbasedAuthentication` in `/etc/ssh/sshd_config`).

O seguinte inicia uma ligação `ssh(1)` a partir de um cliente.

ficheiro de configuração	descrição do ficheiro de configuração
/etc/ssh/ssh_config	Predefinições do cliente SSH, veja <code>ssh_config(5)</code>
/etc/ssh/sshd_config	Predefinições do servidor SSH, veja <code>sshd_config(5)</code>
~/.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	secret SSH-2 <i>key-type-name</i> key such as <code>ecdsa</code> , <code>ed25519</code> , ... of the user

Tabela 6.9: Lista de ficheiros de configuração do SSH

comando	descrição
<code>ssh nome_utilizador@máquina.domínio.externo</code>	ligar com modo predefinido
<code>ssh -v nome_utilizador@máquina.domínio.externo</code>	ligar com modo predefinido com mensagens de depuração
<code>ssh -o PreferredAuthentications=password username@hostname.domain.ext</code>	força o uso de palavra-passe com SSH versão 2
<code>ssh -t username@hostname.domain.ext passwd</code>	run <code>passwd</code> program to update password on a remote host

Tabela 6.10: Lista de exemplos de arranque do cliente SSH

6.3.2 User name on the remote host

If you use the same user name on the local and the remote host, you can eliminate typing “username@”.

Even if you use different user name on the local and the remote host, you can eliminate it using “~/.ssh/config”. For [Debian Salsa service](#) with account name “foo-guest”, you set “~/.ssh/config” to contain the following.

```
Host salsa.debian.org people.debian.org
User foo-guest
```

6.3.3 Ligar sem palavras-passe remotas

One can avoid having to remember passwords for remote systems by using “PubkeyAuthentication” (SSH-2 protocol).

On the remote system, set the respective entries, “PubkeyAuthentication yes”, in “/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"
```

You can add options to the entries in “~/.ssh/authorized_keys” to limit hosts and to run specific commands. See `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 (http://www.chiark.greenend.org.uk/~sgtatham/putty/) (GPL)
Windows (cygwin)	SSH em cygwin (http://www.cygwin.com/) (GPL)
Macintosh Clássico	macSSH (http://www.macssh.com/) (GPL)
Mac OS X	OpenSSH; use SSH na aplicação de Terminal (GPL)

Tabela 6.11: 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 Sending a mail from a remote host

If you have an SSH shell account on a server with proper DNS settings, you can send a mail generated on your workstation as an email genuinely sent from the remote server.

```
$ ssh username@example.org /usr/sbin/sendmail -bm -ti -f "username@example.org" < mail_data ↵
.txt
```

6.3.7 Reencaminhamento de portas 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 `AllowTcpForwarding` 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 "chave_da_máquina" 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

In the old Unix-like system, the BSD [Line printer daemon \(lpd\)](#) was the standard and the standard print out format of the classic free software was [PostScript \(PS\)](#). Some filter system was used along with [Ghostscript](#) to enable printing to the non-PostScript printer. See Secção [11.4.1](#).

In the modern Debian system, the [Common UNIX Printing System \(CUPS\)](#) is the de facto standard and the standard print out format of the modern free software is [Portable Document Format \(PDF\)](#).

The CUPS uses [Internet Printing Protocol \(IPP\)](#). The IPP is now supported by other OSs such as Windows XP and Mac OS X and has became new cross-platform de facto standard for remote printing with bi-directional communication capability.

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:3, I:4	362	impressora (515)	BSD lpr/lpd (daemon de impressora de linha)
lprng	V:0, I:1	3064	, ,	, , (Avançado)
cups	V:129, I:376	1134	IPP (631)	Servidor CUPS de Impressão em Internet
cups-client	V:139, I:434	493	, ,	Comandos de impressão do System V para o CUPS: <code>lp(1)</code> , <code>lpstat(1)</code> , <code>lpoptions(1)</code> , <code>cancel(1)</code> , <code>lpmove(8)</code> , <code>lpinfo(8)</code> , <code>lpadmin(8)</code> , ...
cups-bsd	V:35, I:361	122	, ,	comandos de impressão BSD para o CUPS: <code>lpr(1)</code> , <code>lpq(1)</code> , <code>lprm(1)</code> , <code>lpc(8)</code>
printer-driver-gutenprint	V:55, I:343	941	Não aplicável	Drivers de impressoras para o CUPS

Tabela 6.12: 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:1, I:2	111	TELNET	Servidor TELNET
telnetd-ssl	V:0, I:0	169	, ,	. . (suporte a SSL)
nfs-kernel-server	V:42, I:78	342	NFS	Partilha de ficheiros do Unix
samba	V:105, I:155	16630	SMB	Partilha de ficheiros e impressoras do Windows
netatalk	V:2, I:3	2077	ATP	Partilha de ficheiros e impressoras do Apple/Mac (AppleTalk)
proftpd-basic	V:23, I:31	488	FTP	Descarrega de ficheiros geral
apache2	V:244, I:310	614	HTTP	Servidor web geral
squid	V:12, I:14	8381	, ,	servidor proxy web geral
squid3	V:3, I:9	240	, ,	, ,
bind9	V:52, I:63	1070	DNS	Endereço IP para outras máquinas
isc-dhcp-server	V:18, I:52	1471	DHCP	Endereço IP do próprio cliente

Tabela 6.13: 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 [isc-dhcp-server](#) 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 de alguns ISPs, precisa do cliente [telnet](#) com [TLS/SSL](#) ativo pelos pacotes [telnet-ssl](#) or [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	I:43	16	TCP/IP	Canivete suíço do TCP/IP
openssl	V:814, I:994	1465	SSL	Binário Secure Socket Layer (SSL) e ferramentas criptográficas relacionadas
stunnel4	V:6, I:16	520	, ,	Wrapper SSL universal
telnet	V:61, I:901	163	TELNET	Cliente TELNET
telnet-ssl	V:0, I:3	209	, ,	. . (suporte a SSL)
nfs-common	V:182, I:323	768	NFS	Partilha de ficheiros do Unix
smbclient	V:16, I:174	2017	SMB	Cliente de partilha de ficheiros e impressoras do MS Windows
cifs-utils	V:29, I:123	307	, ,	Comandos de montar e desmontar para ficheiros remotos do MS Windows
ftp	V:16, I:251	137	FTP	Cliente FTP
lftp	V:6, I:40	2255	, ,	, ,
ncftp	V:3, I:21	1339	, ,	Cliente FTP de écran completo
wget	V:255, I:987	3477	HTTP e FTP	descarregador de web
curl	V:160, I:575	427	, ,	, ,
axel	V:0, I:4	216	, ,	acelerador de descarregas
aria2	V:2, I:19	1854	, ,	acelerador de descarregas com suporte de BitTorrent e Metalink
bind9-host	V:176, I:946	370	DNS	host(1) do bind9, "Prioridade: standard"
dnsutils	V:59, I:490	261	, ,	dig(1) do bind, "Prioridade: standard"
isc-dhcp-client	V:222, I:980	686	DHCP	obter endereço IP
ldap-utils	V:12, I:74	719	LDAP	obter dados de um servidor LDAP

Tabela 6.14: 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	Multipurpose Internet Mail Extensions (MIME)
rfc819	serviço DNS
rfc2616	serviço HTTP
rfc2396	definição URI

Tabela 6.15: Lista de RFCs populares

Capítulo 7

GUI System

7.1 GUI desktop environment

There are several choices for the full featured [GUI](#) desktop environment on the Debian system.

task package	popcon	tamanho	descrição
task-gnome-desktop	1:166	9	GNOME desktop environment
task-xfce-desktop	1:102	9	Xfce desktop environment
task-kde-desktop	1:65	6	KDE Plasma desktop environment
task-mate-desktop	1:47	9	MATE desktop environment
task-cinnamon-desktop	1:36	9	Cinnamon desktop environment
task-lxde-desktop	1:33	9	LXDE desktop environment
task-lxqt-desktop	1:13	9	LXQt desktop environment
task-gnome-flashback-desktop	1:0	6	GNOME Flashback desktop environment

Tabela 7.1: List of desktop environment

Dica

Dependency packages selected by a task metapackage may be out of sync with the latest package transition state under the Debian unstable/testing environment. For `task-gnome-desktop`, you may need to adjust package selections as follows:

- Start `aptitude(8)` as `sudo aptitude -u`.
 - Move cursor to "Tasks" and press "Enter".
 - Move cursor to "End-user" press "Enter".
 - Move cursor to "GNOME" press "Enter".
 - Move cursor to `task-gnome-desktop` and press "Enter".
 - Move cursor to "Depends" and press "m" (manually selected).
 - Move cursor to "Recommends" and press "m" (manually selected).
 - Move cursor to `task-gnome-desktop` and press "-". (drop)
 - Adjust selected packages while dropping problematic ones causing package conflicts.
 - Press "g" to start install.
-

This chapter will focus mostly on the default desktop environment of Debian: `task-gnome-desktop` offering [GNOME](#) on [wayland](#).

7.2 GUI communication protocol

GUI communication protocol used on the GNOME desktop can be:

- [Wayland \(display server protocol\)](#) (native)
- [X Window System core protocol](#) (via `xwayland`)

Please check [freedesktop.org site for how Wayland architecture is different from X Window architecture](#).

From user's perspective, differences can be colloquially summarized as:

- Wayland is a same-host GUI communication protocol: new, simpler, faster, no `setuid root` binary
- X Window is a network-capable GUI communication protocol: traditional, complex, slower, `setuid root` binary

For applications using Wayland protocol, the access to their display contents from a remote host is supported by the [VNC](#) or [RDP](#). See [Seção 7.7](#)

Modern X servers have [the MIT Shared Memory Extension](#) and communicate with their local X clients using the local shared memory. This bypasses the network transparent [Xlib](#) interprocess communication channel and gains performance. This situation was the [background](#) of creating Wayland as a local-only GUI communication protocol.

Using the `xeyes` program started from the GNOME terminal, you can check GUI communication protocol used by each GUI application.

```
$ xeyes
```

- If the mouse cursor is on an application such as "GNOME terminal" which uses Wayland display server protocol, eyes don't move with the mouse cursor.
-

- If the mouse cursor is on an application such as "xterm" which uses X Window System core protocol, eyes move with the mouse cursor exposing not-so-isolated nature of X Window architecture.

As of April 2021, many popular GUI applications such as GNOME and [LibreOffice \(LO\)](#) applications have been migrated to the Wayland display server protocol. I see xterm, gitk, chromium, firefox, gimp, dia, and KDE applications still use X Window System core protocol.

Nota

For both the xwayland on Wayland or the native X Window System, the old X server configuration file `/etc/X11/xorg.conf` shouldn't exist on the system. The graphics and input devices are now configured by the kernel with [DRM](#), [KMS](#), and [udev](#). The native X server has been rewritten to use them. See "[modeb default video mode support](#)" in the Linux kernel documentation.

7.3 GUI infrastructure

Here are notable GUI infrastructure packages for the GNOME on Wayland environment.

pacote	popcon	tamanho do pacote	descrição
mutter	V:7, I:204	208	GNOME's mutter window manager [auto]
xwayland	V:137, I:227	4572	An X server running on top of wayland [auto]
gnome-remote-desktop	V:14, I:23	207	Remote desktop daemon for GNOME using PipeWire [auto]
gnome-tweaks	V:14, I:169	1264	Advanced configuration settings for GNOME

Tabela 7.2: List of notable GUI infrastructure packages

Here, "**[auto]**" means that these packages are automatically installed when `task-gnome-desktop` is installed.

Dica

`gnome-tweaks` is the indispensable configuration utility. For example:

- You can force "Over-Amplification" of sound volume from "General".
 - You can force "Caps" to become "Esc" from "Keyboard & Mouse" -> "Keyboard" -> "Additional Layout Option".
-

7.4 GUI applications

Many useful GUI applications are available on Debian now. Installing software packages such as `scribus` (KDE) on GNOME desktop environment are quite acceptable since corresponding functionality is not available under GNOME desktop environment. But installing too many packages with duplicated functionalities may clutter your system.

Here is a list of GUI applications which caught my eyes.

7.5 Fonts

Many useful scalable fonts are available for users on Debian. User's concern is how to avoid redundancy and how to configure parts of installed fonts to be disabled. Otherwise, useless font choices may clutter your GUI application menus.

Debian system uses [FreeType](#) 2.0 library to rasterise many scalable font formats for screen and print:

pacote	popcon	tamanho do pa- cote	tipo	descrição
evolution	V:29, I:219	475	GNOME	Gestor de Informações Pessoais (groupware e email)
thunderbird	V:57, I:134	168690	GTK	Email client (Mozilla Thunderbird)
kontact	V:1, I:14	2152	KDE	Gestor de Informações Pessoais (groupware e email)
libreoffice-writer	V:167, I:423	39325	LO	processador de texto
abiword	V:1, I:11	5133	GNOME	processador de texto
calligrawords	V:0, I:7	5717	KDE	processador de texto
scribus	V:1, I:22	30370	KDE	desktop publishing editor to edit PDF files
glabels	V:0, I:4	1326	GNOME	editor de etiquetas
libreoffice-calc	V:166, I:418	32969	LO	folha de cálculo
gnumeric	V:5, I:20	9933	GNOME	folha de cálculo
calligrasheets	V:0, I:5	10890	KDE	folha de cálculo
libreoffice-impress	V:149, I:415	9930	LO	apresentação
calligrastage	V:0, I:5	5102	KDE	apresentação
libreoffice-base	V:116, I:289	7465	LO	gestão de base de dados
kexi	V:0, I:2	7576	KDE	gestão de base de dados
libreoffice-draw	V:150, I:416	14596	LO	editor de gráficos vectoriais (desenho)
inkscape	V:42, I:188	84892	GNOME	editor de gráficos vectoriais (desenho)
karbon	V:0, I:6	3473	KDE	editor de gráficos vectoriais (desenho)
dia	V:4, I:30	3727	GTK	editor de fluxogramas e diagramas
gimp	V:58, I:315	19706	GTK	editor de gráficos bitmap (pintura)
shotwell	V:16, I:217	6451	GTK	organizador de fotos digitais
digikam	V:2, I:12	2646	KDE	organizador de fotos digitais
darktable	V:6, I:17	22902	GTK	lighttable and darkroom for photographers
planner	V:0, I:4	1146	GNOME	gestão de projectos
calligraplan	V:0, I:2	18517	KDE	gestão de projectos
gnucash	V:3, I:11	32304	GNOME	contas pessoais
homebank	V:0, I:2	1114	GTK	contas pessoais
lilypond	V:0, I:8	7337	-	music typesetter
kmymoney	V:0, I:2	12036	KDE	contas pessoais
librecad	V:3, I:18	7893	Qt-app	computer-aided design (CAD) system (2D)
freecad	I:19	53	Qt-app	computer-aided design (CAD) system (3D)
kicad	V:2, I:15	98707	GTK	electronic schematic and PCB design software
xsane	V:14, I:165	2346	GTK	frontend para digitalizador (scanner)
libreoffice-math	V:144, I:419	1963	LO	editor de fórmulas/equações matemáticas
calibre	V:8, I:35	55048	KDE	conversor de e-books e gestor de biblioteca
fbreader	V:1, I:14	3074	GTK	leitor de e-book
evince	V:105, I:313	956	GNOME	visualizador de documentos (pdf)
okular	V:43, I:115	14646	KDE	visualizador de documentos (pdf)
x11-apps	V:27, I:454	2437	pure X-app	xeyes (1), etc.
x11-utils	V:171, I:581	712	pure X-app	xev (1), xwininfo (1)etc.

Tabela 7.3: List of notable GUI applications

- [Type 1 \(PostScript\) fonts](#) which use cubic [Bézier curves](#) (almost obsolete format)
- [TrueType fonts](#) which use quadratic [Bézier curves](#) (good choice format)
- [OpenType fonts](#) which use cubic [Bézier curves](#) (best choice format)

7.5.1 Fontes (tipos de letra) básicas

The following table is compiled in the hope to help users to chose appropriate scalable fonts with clear understanding of the metric compatibility and the glyph coverage. Most fonts cover all Latin fonts, Greek, and Cyril character glyphs. The final choice of activated fonts can also be affected by your aesthetics. These fonts can be used for the screen display or for the paper printing.

pacote	popcon	tamanho	sans	serif	mono	note on font
fonts-cantarell	V:37, I:273	836	59	-	-	Cantarell (GNOME 3, display)
fonts-noto	I:133	35	61	63	40	Noto fonts (Google, multi-lingual with CJK)
fonts-dejavu	I:462	39	58	68	40	DejaVu (GNOME 2, MCM: Verdana , extended Bitstream Vera)
fonts-liberation2	V:48, I:333	4261	56	60	40	Liberation fonts for LibreOffice (Red Hat, MCMATC)
fonts-croscore	V:10, I:47	5278	56	60	40	Chrome OS: Arimo, Tinos and Cousine (Google, MCMATC)
fonts-crosextra-carlito	I:276	2732	57	-	-	Chrome OS: Carlito (Google, MCM: Calibri)
fonts-crosextra-caladea	I:275	258	-	55	-	Chrome OS: Caladea (Google, MCM: Cambria) (Latin only)
fonts-freefont-ttf	V:55, I:257	6656	57	59	40	GNU FreeFont (extended URW Nimbus)
fonts-quicksand	I:358	392	56	-	-	Debian task-desktop, Quicksand (display, Latin only)
fonts-hack	V:9, I:89	2508	-	-	40 P	A typeface designed for source code Hack (Facebook)
fonts-sil-gentiumplus	I:39	5756	-	54	-	Gentium SIL
fonts-sil-charis	V:2, I:12	8900	-	59	-	Charis SIL
fonts-urw-base35	V:46, I:94	11095	56	60	40	URW Nimbus (Sans , Roman No. 9 L , Mono L , MCAHTC)
fonts-ubuntu	V:1, I:5	4103	58	-	33 P	Ubuntu fonts (display)
fonts-terminus	V:0, I:1	450	-	-	33	Cool retro terminal fonts
ttf-mscorefonts-installer	V:1, I:61	92	56?	60	40	Downloader of Microsoft non-free fonts (see below)

Tabela 7.4: List of notable [TrueType](#) and [OpenType](#) fonts

Here:

- "MCM" stands for "metric compatible with fonts provided by Microsoft"

- "MCMATC" stands for "metric compatible with fonts provided by Microsoft: [Arial](#), [Times New Roman](#), [Courier New](#)"
- "MCAHTC" stands for "metric compatible with fonts provided by [Adobe](#): Helvetica, Times, Courier"
- Numbers in font type columns stands for the rough relative "M" width for the same point size font.
- "P" in mono font type columns stands for its usability for programming having clearly distinguishable "0"/"O" and "1"/"l"/"I".
- The `ttf-mscorefonts-installer` package downloads Microsoft's "[Core fonts for the Web](#)" and installs [Arial](#), [Times New Roman](#), [Courier New](#), [Verdana](#), These installed font data are non-free data.

Many free Latin fonts have their lineage traced to [URW Nimbus](#) family or [Bitstream Vera](#).

Dica

If your locale needs fonts not covered well by the above fonts, please use aptitude to check under task packages listed under "Tasks" -> "Localization". The font packages listed as "Depends:" or "Recommends:" in the localization task packages are the primary candidates.

7.5.2 Font rasterization

Debian uses [FreeType](#) to rasterize fonts. Its font choice infrastructure is provided by the [Fontconfig](#) font configuration library.

pacote	popcon	tamanho	descrição
libfreetype6	V:432, I:995	896	FreeType font rasterization library
libfontconfig1	V:432, I:860	526	Fontconfig font configuration library
fontconfig	V:353, I:757	583	<code>fc - *</code> : CLI commands for Fontconfig
font-manager	V:1, I:9	1015	Font Manager : GUI command for Fontconfig
nautilus-font-manager	V:0, I:0	37	Nautilus extension for Font Manager

Tabela 7.5: List of notable font environment and related packages

Dica

Some font packages such as `fonts-noto*` install too many fonts. You may also want to keep some font packages installed but disabled under the normal use situation. The multiple [glyphs](#) are expected for some [Unicode](#) code points due to [Han unification](#) and unwanted glyphs may be chosen by the unconfigured [Fontconfig](#) library. One of the most annoying case is "U+3001 IDEOGRAPHIC COMMA" and "U+3002 IDEOGRAPHIC FULL STOP" among CJK countries. You can avoid this problematic situation easily by configuring font availability using [Font Manager](#) GUI ([font-manager](#)).

You can list font configuration state from the command line, too.

- `"fc-match(1)"` for fontconfig font default
- `"fc-list(1)"` for available fontconfig fonts

You can configure font configuration state from the text editor but this is non-trivial. See `fonts.conf(5)`.

7.6 Sandbox

Many mostly GUI applications on Linux are available in binary formats from non-Debian sources.

- [AppImage -- Linux apps that run anywhere](#)
- [FLATHUB -- Apps for Linux, right here](#)
- [snapcraft -- The app store for Linux](#)



Atenção

Binaries from these sites may include proprietary non-free software packages.

There is some *raison d'être* for these binary format distributions for Free Software aficionados using Debian since these can accommodate clean set of libraries used for each application by the respective upstream developer independent of the ones provided by Debian.

The inherent risk of running external binaries can be reduced by using the [sandbox environment](#) which leverages modern Linux security features (see Seção [4.7.4](#)).

- For binaries from AppImage and some upstream sites, run them in [firejail](#) with [manual configuration](#).
- For binaries from FLATHUB, run them in [Flatpak](#) . (No manual configuration required.)
- For binaries from snapcraft, run them in [Snap](#) . (No manual configuration required. Compatible with daemon programs.)

The `xdg-desktop-portal` package provides a standardized API to common desktop features. See [xdg-desktop-portal \(flatpak\)](#) and [xdg-desktop-portal \(snap\)](#)

pacote	popcon	tamanho	descrição
flatpak	V:31, I:35	6562	Flatpak application deployment framework for desktop apps
gnome-software-plugin-flatpak	V:4, I:14	335	Flatpak support for GNOME Software
snapd	V:53, I:58	55035	Daemon and tooling that enable snap packages
gnome-software-plugin-snap	V:0, I:2	237	Snap support for GNOME Software
xdg-desktop-portal	V:125, I:209	1425	desktop integration portal for Flatpak and Snap
xdg-desktop-portal-gtk	V:71, I:205	921	xdg-desktop-portal backend for gtk (GNOME)
xdg-desktop-portal-kde	V:1, I:4	842	xdg-desktop-portal backend for Qt (KDE)
xdg-desktop-portal-wlr	V:0, I:0	83	xdg-desktop-portal backend for wlroots (Wayland)
firejail	V:1, I:5	1370	a SUID security sandbox program firejail for use with AppImage

Tabela 7.6: List of notable sandbox environment and related packages

This sandbox environment technology is very much like apps on smart phone OS where apps are executed under controlled resource accesses.

Some large GUI applications such as web browsers on Debian also use sandbox environment technology internally to make them more secure.

pacote	popcon	tamanho	protocols	descrição
gnome-remote-desktop	V:14, I:23	207	RDP, RFB (VNC)	GNOME Remote Desktop server
vinagre	V:9, I:222	4249	RDP, RFB (VNC), SPICE, SSH	Vinagre: GNOME remote desktop client
remmina	V:12, I:47	550	RDP, RFB (VNC), SPICE, SSH, ...	Remmina: GTK remote desktop client
krdc	V:2, I:21	3538	RDP, RFB (VNC)	KRDC: KDE remote desktop client
guacd	V:0, I:0	83	RDP, RFB (VNC), SSH / HTML5	Apache Guacamole: clientless remote desktop gateway (HTML5)
virt-viewer	V:4, I:51	1545	RFB (VNC), SPICE	Virtual Machine Manager's GUI display client of guest OS

Tabela 7.7: List of notable remote access server

7.7 Remote desktop

7.8 X server connection

There are several ways to connect from an application on a remote host to the X server including [xwayland](#) on the local host.

pacote	popcon	tamanho	comando	descrição
openssh-server	V:701, I:844	1567	sshd with option X11-forwarding	SSH server (secure)
openssh-client	V:809, I:997	4298	ssh -X	SSH client (secure)
xauth	V:155, I:951	86	xauth	X authority file utility
x11-xserver-utils	V:272, I:516	511	xhost	server access control for X

Tabela 7.8: Lista de métodos de ligação ao servidor X

7.8.1 X server local connection

Access to the local X server by the local applications which use X core protocol can be locally connected through a local UNIX domain socket. This can be authorized by the authority file holding [access cookie](#). The authority file location is identified by the "\$XAUTHORITY" environment variable and X display is identified by the "\$DISPLAY" environment variable. Since these are normally set automatically, no special action is needed, e.g. "gitk" as the following.

```
username $ gitk
```

Nota

For [xwayland](#), XAUTHORITY holds value like `"/run/user/1000/.mutter-xwaylandauth.YVSU30"`.

7.8.2 X server remote connection

Access to the local X server display from the remote applications which use X core protocol is supported by using the X11 forwarding feature.

- Open an `gnome-terminal` on the local host.
- Run `ssh(1)` with `-X` option to establish a connection with the remote site as the following.

```
localname @ localhost $ ssh -q -X loginname@remotehost.domain
Password:
```

- Run an X application command, e.g. `"gitk"`, on the remote site as the following.

```
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.

See Secção [6.3](#) for SSH/SSHD.



Atenção

A remote [TCP/IP](#) to the X server is disabled by default on the Debian system for security reasons. Don't enable them by simply setting `"xhost +"` nor by enabling [XDMCP connection](#), if you can avoid it.

7.8.3 X server chroot connection

Access to the X server by the applications which use X core protocol and run on the same host but in an environment such as `chroot` where the authority file is not accessible, can be authorized securely with `xhost` by using the [User-based access](#), e.g. `"gitk"` as the following.

```
username $ xhost + si:localuser:root ; sudo chroot /path/to
# cd /src
# gitk
# exit
username $ xhost -
```

7.9 Área de transferência (Clipboard)

For clipping text to clipboard, see Secção [1.4.4](#).

For clipping graphics to clipboard, see Secção [11.6](#).

Some CLI commands can manipulate character clipboard (PRIMARY and CLIPBOARD), too.

pacote	popcon	tamanho do pa- cote	alvo	descrição
xsel	V:9, I:43	59	X	command line interface to X selections (clipboard)
xclip	V:10, I:51	64	X	command line interface to X selections (clipboard)
wl-clipboard	V:0, I:1	129	Wayland	wl-copy wl-paste : command line interface to Wayland clipboard
gpm	V:11, I:16	540	Consola do Linux	a daemon that captures mouse events on Linux console

Tabela 7.9: List of programs related to manipulating character clipboard

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 locais.
- Localization (L10N): Tornar o software útil num locale específico.

Dica

There are 17, 18, or 10 letters between "m" and "n", "i" and "n", or "l" and "n" in multilingualization, internationalization, and localization which correspond to M17N, I18N, and L10N. See [Introduction to i18n](#) for details.

8.1 O locale

The behavior of programs supporting internationalization are configured by the environment variable "\$LANG" to support localization. Actual support of locale dependent features by the `libc` library requires to install `locales` or `locales-all` packages. The `locales` package requires to be initialized properly.

If neither `locales` or `locales-all` package are installed, support of locale features are lost and system uses US English messages and handles data as **ASCII**. This behavior is the same way as "\$LANG" is set by "LANG=", "LANG=C", or "LANG=POSIX".

The modern software such as GNOME and KDE are multilingualized. They are internationalized by making them handle [UTF-8](#) data and localized by providing their translated messages through the `gettext(1)` infrastructure. Translated messages may be provided as separate localization packages.

The current Debian desktop GUI system normally sets the locale under GUI environment as "LANG=xx_YY.UTF-8". Here, "xx" is [ISO 639 language codes](#) and "YY" is [ISO 3166 country codes](#). These values are set by the desktop configuration GUI dialogue and change the program behavior. See [Secção 1.5.2](#)

8.1.1 Fundamentos para o locale UTF-8

The simplest representation of the text data is **ASCII** which is sufficient for English and uses less than 127 characters (representable with 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'
```

In order to support more characters, many character sets and encoding systems have been used to support many languages (see Tabela 11.2).

[Unicode](#) character set can represent practically all characters known to human with 21 bit code point range (i.e., 0 to 10FFFF in hexadecimal notation).

Text encoding system [UTF-8](#) fits Unicode code points into a sensible 8 bit data stream mostly compatible with the ASCII data processing system. This makes **UTF-8** the modern preferred choice. **UTF** stands for Unicode Transformation Format. When [ASCII](#) plain text data is converted to [UTF-8](#) one, it has exactly the same content and size as the original ASCII one. So you loose nothing by deploying UTF-8 locale.

Under [UTF-8](#) locale with the compatible application program, you can display and edit any foreign language text data as long as required fonts and input methods are installed and enabled. For example under "`LANG=fr_FR.UTF-8`" locale, `gedit(1)` (text editor for the GNOME Desktop) can display and edit Chinese character text data while presenting menus in French.

Dica

Both the new standard "`en_US.UTF-8`" locale and the old standard "`C`"/"`POSIX`" locale use the standard US English message, they have subtle differences in sorting order etc. If you want to handle not only ASCII characters but also handle all UTF-8 encoded characters gracefully while maintaining the old "`C`" local behavior, use the non-standard "`C.UTF-8`" locale on Debian.

Nota

Alguns programas consomem mais memória após suportarem l18n. Isto é porque estão codificados para usar [UTF-32\(UCS4\)](#) internamente para suportar Unicode para otimização de velocidade e consomem 4 bytes por cada caractere ASCII independentemente do locale selecionado. Mais uma vez, não perde nada ao implantar o locale UTF-8.

8.1.2 A reconfiguração do locale

In order for the system to access a particular locale, the locale data must be compiled from the locale database.

The `locales` package does **not** come with pre-compiled locale data. You need to configure it as:

```
# dpkg-reconfigure locales
```

This process involves 2 steps.

1. Select all required locale data to be compiled into the binary form. (Please make sure to include at least one UTF-8 locale)
2. Set the system wide default locale value by creating "`/etc/default/locale`" for use by PAM (see Secção 4.5).

The system wide default locale value set in "`/etc/default/locale`" may be overridden by the GUI configuration for GUI applications.

Nota

Actual traditional encoding system can be identified by "`/usr/share/i18n/SUPPORTED`". Thus, the "`LANG=en_US`" is "`LANG=en_US.ISO-8859-1`".

The `locales-all` package comes with all locale data pre-compiled but doesn't creating "`/etc/default/locale`".

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 caneta de memória 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 caneta de memória 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
```

GNU gettext can accommodate priority list of translation languages with `$LANGUAGE` environment variable. For example:

```
$ export LANGUAGE="pt:pt_BR:es:it:fr"
```

For more, see `info gettext` and read the section "The LANGUAGE variable".

8.1.5 Efeitos do locale

A ordem de ordenação do caracteres com o `sort(1)` é afectada pela escolha de linguagem do locale. Os locales Espanhol e Inglês ordenam de forma diferente.

O formato de data do `ls(1)` é afectado pelo locale. Os formatos de data de `"LANG=C ls -l"` e `"LANG=en_US.UTF-8"` são diferentes (veja Secção 9.3.4).

As pontuações numéricas são diferentes para os locales. Por exemplo, em locale Inglês, mil ponto um é mostrado como `"1,000.1"` enquanto em locale Alemão é mostrado como `1.000,1`. Pode ver esta diferença num programa de folha de cálculo.

Each detail feature of `"$LANG"` environment variable may be overridden by setting `"$LC_*` variables. These environment variables can be overridden again by setting `"$LC_ALL"` variable. See `locale(7)` manpage for the details. Unless you have strong reason to create complicated configuration, please stay away from them and use only `"$LANG"` variable set to one of the UTF-8 locales.

8.2 A entrada do teclado

8.2.1 The keyboard input for Linux console and X Window

O sistema Debian pode ser configurado para funcionar com muitas disposições internacionais de teclado a usar os pacotes `keyboard-configuration` e `console-setup`.

```
# dpkg-reconfigure keyboard-configuration
# dpkg-reconfigure console-setup
```

For the Linux console and the X Window system, this updates configuration parameters in `/etc/default/keyboard` and `/etc/default/console-setup`. This also configures the Linux console font. Many non-ASCII characters including accented characters used by many European languages can be made available with [dead key](#), [AltGr key](#), and [compose key](#).

8.2.2 The keyboard input for Wayland

For GNOME on Wayland desktop system, Seção 8.2.1 can't support non-English European languages. [IBus](#) was made to support not only Asian languages but also European languages. The package dependency of GNOME Desktop Environment recommends "ibus" via "gnome-shell". The code of "ibus" has been updated to integrate `setxkbmap` and XKB option functionalities. You need to configure `ibus` from "GNOME Settings" or "GNOME Tweaks" for the multilingualized keyboard input.

Nota

If `ibus` is active, your classic X keyboard configuration by the `setxkbmap` may be overridden by `ibus` even under classic X-based desktop environment. You can disable installed `ibus` using `im-config` to set input method to "None". For more, see [Debian Wiki on keyboard](#).

8.2.3 O suporte a método de entrada com IBus

Since GNOME Desktop Environment recommends "ibus" via "gnome-shell", "ibus" is the good choice for input method. Entrada multilingue para a aplicação é processada como:

```
Keyboard                                Application
|                                     ^
|                                     |
+--> Linux kernel -> Input method (ibus) -> Gtk, Qt, X, Wayland
      +-- Engine--+
```

The list of IBus and its engine packages are the following.

Nota

For Chinese, "fcitx" may be an alternative input method framework. For Emacs aficionados, "uim" may be an alternative. Either cases, you may need to do extra manual configuration with `im-config`. Some old classic [input methods](#) such as "kinput2" may still exist in Debian repository but are not recommended for the modern environment.

8.2.4 Um exemplo para Japonês

I find the Japanese input method started under English environment ("`en_US.UTF-8`") very useful. Here is how I did this with IBus for GNOME on Wayland:

1. Install the Japanese input tool package `ibus-mozc` (or `ibus-anthy`) with its recommended packages such as `im-config`.
-

pacote	popcon	tamanho	locale suportado
ibus	V:82, I:99	1584	estrutura de método de entrada que usa dbus
ibus-mozc	V:1, I:2	1006	Japonês
ibus-anthy	V:1, I:1	8790	, ,
ibus-kkc	V:0, I:0	214	, ,
ibus-skk	V:0, I:0	244	, ,
ibus-pinyin	V:0, I:1	1434	Chinês (para zh_CN)
ibus-chewing	V:0, I:0	415	, , (para zh_TW)
ibus-hangul	V:0, I:1	288	Coreano
ibus-table	V:0, I:1	1809	motor de tabela para IBus
ibus-table-thai	I:0	47	Thai
ibus-unikey	V:0, I:0	318	Vietnamita
ibus-m17n	V:0, I:1	187	Multilíngue: Indiano, Árabe e outros

Tabela 8.1: List of IBus and its engine packages

2. Select "Settings" → "Keyboard" → "Input Sources" → click "+" in "Input Sources" → "Japanese" → "Japanese mozc (or anthy)" and click "Add" if it hasn't been activated.
3. You may chose as many input sources.
4. Voltar a fazer login na conta do utilizador.
5. Setup each input source by right clicking the GUI toolbar icon.
6. Alterne entre as fontes de entrada instaladas com SUPER-ESPAÇO. (SUPER é normalmente a Tecla Windows.)

Dica

If you wish to have access to alphabet only keyboard environment with the physical Japanese keyboard on which shift-2 has " (double quotation mark) engraved, you select "Japanese" in the above procedure. You can enter Japanese using "Japanese mozc (or anthy)" with physical "US" keyboard on which shift-2 has @ (at mark) engraved.

- The GUI menu entry for `im-config(8)` is "Input method".
- Alternatively, execute "`im-config`" from user's shell.
- 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

Linux console can only display limited characters. (You need to use special terminal program such as `jfbterm(1)` to display non-European languages on the non-GUI console.)

GUI environment (Capítulo 7) can display any characters in the UTF-8 as long as required fonts are installed and enabled. (The encoding of the original font data is taken care and transparent to the user.)

8.4 Caracteres Asiáticos de Leste de Altura Ambígua

Em localizações ao leste da Ásia, o desenhar de caixa, os caracteres Gregos e Cirílicos podem aparecer mais largos que o desejado e causarem o desalinhamento da saída do terminal (veja [Unicode Standard Annex #11](#)).

Pode contornar este problema:

- `gnome-terminal`: Preferences → Profiles → *Profile name* → Compatibility → Ambiguous-wide characters → Narrow
- `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 The console tips

There are some utility programs to help your console activities.

pacote	popcon	tamanho	descrição
mc	V:55, I:234	1492	See Secção 1.3
bsdutils	V:708, I:999	394	<code>script</code> command to make typescript of terminal session
screen	V:114, I:283	1014	terminal multiplexador com emulação de terminal VT100/ANSI
tmux	V:35, I:143	830	multiplexador alternativo de terminal (Use "Control-B" em vez disto)
fzf	V:2, I:7	2637	fuzzy text finder
fzy	V:0, I:0	54	fuzzy text finder
rlwrap	V:1, I:20	297	readline feature command line wrapper
ledit	V:0, I:14	306	readline feature command line wrapper
rlfe	V:0, I:0	49	readline feature command line wrapper

Tabela 9.1: List of programs to support console activities

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
```

There are alternative methods to record the shell activities:

- Use `tee` (usable during the boot process in the `initramfs`):

```
$ sh -i 2>&1 | tee typescript
```

- Use `gnome-terminal` with the extend line buffer for scrollbar.
- Use `screen` with `^A H` (see Secção 9.1.2) to perform recording of console.
- Use `vim` with `":terminal"` to enter the terminal mode. Use `Ctrl-W N` to exit from terminal mode to normal mode. Use `":w typescript"` to write the buffer to a file.
- Use `emacs` with `M-x shell`, `M-x eshell`, or `M-x term` to enter recording console. Use `C-x C-w` to write the buffer to a file.

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.

See `tmux(1)` for functionalities of the alternative command.

9.1.3 Navigating around directories

In Secção 1.4.2, 2 tips to allow quick navigation around directories are described: `$CDPATH` and `mc`.

If you use fuzzy text filter program, you can do without typing the exact path. For `fzf`, include following in `~/ .bashrc`.

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

```
FZF_KEYBINDINGS_PATH=/usr/share/doc/fzf/examples/key-bindings.bash
if [ -f $FZF_KEYBINDINGS_PATH ]; then
  . $FZF_KEYBINDINGS_PATH
fi
FZF_COMPLETION_PATH=/usr/share/doc/fzf/examples/completion.bash
if [ -f $FZF_COMPLETION_PATH ]; then
  . $FZF_COMPLETION_PATH
fi
```

For example:

- You can jump to a very deep subdirectory with minimal efforts. You first type `"cd **"` and press **Tab**. Then you will be prompted with candidate paths. Typing in partial path strings, e.g., `s/d/b foo`, will narrow down candidate paths. You select the path to be used by **cd** with cursor and return keys.
- You can select a command from the command history more efficiently with minimal efforts. You press **Ctrl-R** at the command prompt. Then you will be prompted with candidate commands. Typing in partial command strings, e.g., `vim d`, will narrow down candidates. You select the one to be used with cursor and return keys.

9.1.4 Readline wrapper

Some commands such as `/usr/bin/dash` which lacks command line history editing capability can add such functionality transparently by running under `rlwrap` or its equivalents.

```
$ rlwrap dash -i
```

This provides convenient platform to test subtle points for `dash` with friendly `bash`-like environment.

9.2 Personalizar o vim

After you learn basics of `vim`(1) through Secção 1.4.8, please read Bram Moolenaar's "[Seven habits of effective text editing \(2000\)](#)" to understand how `vim` should be used.



Cuidado

Don't try to change the default key bindings without very good reasons.

9.2.1 Customizing vim with internal features

The behavior of `vim` can be changed significantly by enabling its internal features through the Ex-mode commands such as `"set ..."` to set vim options.

These Ex-mode commands can be included in user's vimrc file, traditional `"~/.vimrc"` or git-friendly `"~/.vim/vimrc"`. Here is a very simple example 1:

```
colorscheme murphy          " from /usr/share/vim/vim??/colors/*.vim
filetype plugin indent on   " filetype aware behavior
syntax enable               " Syntax highlight
"set spelllang=en_us        " Spell check language as en_us
"set spell                  " Enable spell check
set autoindent              " Copy indent from current line
set smartindent             " More than autoindent (Drop/Pop after {/})
set nosmarttab              " <Tab>-key always inserts blanks
set backspace=indent,eol,start " Back space through everything
set laststatus=2           " Always show status line
set statusline=%<%f%m%r%h%w%=%y[U+%04B]%2l/%2L=%P,%2c%V
```

9.2.2 Customizing vim with external packages

Simple customization to enable `securemodelines` and classical IDE can be enabled by installing [vim-scripts](#) package and appending the following to user's vimrc file.

```
packadd! securemodelines
packadd! winmanager
let mapleader = ' '
" Toggle paste mode with <SPACE>p
set pastetoggle=<leader>p
" IDE-like UI for files and buffers with <space>w
nnoremap <leader>w          :WMToggle<CR>
" Use safer keys <SPACE>? for moving to another window
nnoremap <leader>h          <C-W>h
nnoremap <leader>j          <C-W>j
nnoremap <leader>k          <C-W>k
nnoremap <leader>l          <C-W>l
```

The new native Vim package system works nicely with `"git"` and `"git submodule"`. One such example configuration can be found at [my git repository: dot-vim](#). This does essentially:

- By using `"git"` and `"git submodule"`, latest external packages, such as `"name"`, are placed into `~/.vim/pack/*/opt/name` and similar.
- By adding `:packadd! name` line to user's vimrc file, these packages are placed on `runtimepath`.
- Vim loads these packages on `runtimepath` during its initialization.
- At the end of its initialization, tags for the installed documents are updated with `"helptags ALL"`.

For more, please start vim with `"vim --startuptime vimstart.log"` to check actual execution sequence and time spent for each step.

Interesting external plugin packages can be found:

- [Vim - the ubiquitous text editor](#) -- The official upstream site of Vim and vim scripts
- [VimAwesome](#) -- The listing of Vim plugins
- [vim-scripts](#) -- Debian package: a collection of vim scripts

It is quite confusing to see too many ways² to manage and load these external packages to vim. Checking the original information

¹More elaborate customization examples: ["Vim Galore"](#), ["sensible.vim"](#), ["#vim Recommendations"](#) ...

²[vim-pathogen](#) was popular.

is the best cure.

key strokes	information
:help package	explanation on the vim package mechanism
:help runtimepath	explanation on the runtimepath mechanism
:version	internal states including candidates for the vimrc file
:echo \$VIM	the environment variable "\$VIM" used to locate the vimrc file
:set runtimepath?	list of directories which will be searched for all runtime support files
:echo \$VIMRUNTIME	the environment variable "\$VIMRUNTIME" used to locate various system provided runtime support files

Tabela 9.3: Information on the initialization of vim

9.3 Gravação de dados e apresentação

9.3.1 O daemon de log

Muitos programas registam as suas atividades sob o diretório `"/var/log/"`.

Veja Secção 3.4 e Secção 3.3.

9.3.2 Analisador de relatório (Log)

Aqui estão analisadores de relatórios notáveis (`"~Gsecurity::log-analyzer"` no `aptitude(8)`).

pacote	popcon	tamanho	descrição
logwatch	V:16, I:18	2265	analisador de log com saída bonita escrito em Perl
fail2ban	V:116, I:129	2092	banir IPs que causam vários erros de autenticação
analog	V:3, I:105	3534	analisador de log do servidor web
awstats	V:9, I:14	6910	analisador de logs de servidor web poderoso e cheio de funcionalidades
sarg	V:2, I:3	843	gerador de relatórios de análises do squid
pflogsumm	V:2, I:4	111	resumidor de entradas do relatório do Postfix
syslog-summary	V:0, I:2	30	resume o conteúdo do um ficheiro de log do syslog
fwlogwatch	V:0, I:0	478	analisador de log da firewall
squidview	V:0, I:1	189	monitoriza e analisa ficheiros access.log do squid
swatch	V:0, I:0	101	visualizador de ficheiros de registo com correspondência de expressões regulares, destaque e hooks
crm114	V:0, I:0	1119	Mutilador de Expressões Regulares Controlável e Filtro de Spam (CRM114)
icmpinfo	V:0, I:0	44	interpretar mensagens ICMP

Tabela 9.4: Lista de analisadores de log do sistema

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.

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

The default display format of time and date by the `"ls -l"` command depends on the **locale** (see Secção 1.2.6 for value). The `"$LANG"` variable is referred first and it can be overridden by the `"$LC_TIME"` or `"$LC_ALL"` exported environment variables.

The actual default display format for each locale depends on the version of the standard C library (the `libc6` package) used. I.e., different releases of Debian had different defaults. For iso-formats, see [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'"`).

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: Display examples of time and date for the `"ls -l"` command with the **time style value**

Dica

You can eliminate typing long option on commandline using command alias (see Secção 1.5.9):

```
alias ls='ls --time-style=+%d.%m.%y %H:%M'
```

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/xterm/c`

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 inspecionar 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	screen
gnome-screenshot	V:27, I:264	1173	Wayland
flameshot	V:5, I:11	1510	Wayland
gimp	V:58, I:315	19706	Wayland + X
x11-apps	V:27, I:454	2437	X
imagemagick	I:373	220	X
scrot	V:7, I:76	123	X

Tabela 9.6: Lista de ferramentas gráficas de manipulação de imagens

pacote	popcon	tamanho	descrição
etckeeper	V:28, I:32	176	armazenar ficheiros de configuração e os seus meta-dados com Git (predefinido), Mercurial , ou Bazaar (novo)
timeshift	V:2, I:4	3158	system restore utility using rsync or BTRFS snapshots
snapper	V:2, I:3	2119	Linux filesystem snapshot management tool

Tabela 9.7: List of packages which can record configuration history

9.3.9 Gravar alterações em ficheiros de configuração

There are specialized tools to record changes in configuration files with help of DVCS and to make system snapshots on [Btrfs](#). You may also think about local script [Secção 10.2.2](#) approach.

9.4 Monitorizar, controlar e iniciar as atividades de programas

As atividades de programas podem ser monitorizadas e controladas a usar ferramentas especiais.

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.

```
# 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.

pacote	popcon	tamanho	descrição
coreutils	V:901, I:999	17478	nice(1) : correr um programa com prioridade de agendamento modificada
bsdutils	V:708, I:999	394	renice(1) : modifica a prioridade de agendamento de um processo em execução
procps	V:742, I:999	1648	"/proc" utilitários de sistema de ficheiros: ps(1) , top(1) , kill(1) , watch(1) , ...
psmisc	V:433, I:831	793	"/proc" utilitários de sistema de ficheiros: killall(1) , fuser(1) , peekfd(1) , pstree(1)
time	V:13, I:249	129	time(1) : corre um programa para reportar as utilizações de recursos do sistema no que respeita a tempo
sysstat	V:168, I:190	1923	sar(1) , iostat(1) , mpstat(1) , ...: ferramentas de performance do sistema para Linux
isag	V:0, I:4	117	Interactive System Activity Grapher para sysstat
lsof	V:393, I:944	451	lsof(8) : lista os ficheiros abertos por um processo em execução a usar a opção "-p"
strace	V:16, I:151	2367	strace(1) : rastreia chamadas e sinais do sistema
ltrace	V:1, I:22	363	ltrace(1) : rastreia chamadas de bibliotecas
xtrace	V:0, I:0	353	xtrace(1) : rastreia a comunicação entre cliente X11 e servidor
powertop	V:8, I:204	662	powertop(1) : informação sobre a utilização do sistema de energia
cron	V:800, I:997	263	corre processos de acordo com uma agenda nos bastidores a partir do daemon cron(8)
anacron	V:399, I:463	104	agenda de comandos tipo cron para sistemas que não funcionam 24 horas por dia
at	V:154, I:278	161	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

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

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	<code>ps aux</code>	mostrar %CPU %MEM
System V	<code>ps -efH</code>	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 Rastrear 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

Use **strace-graph** script found in `/usr/share/doc/strace/examples/` to make a nice tree view

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](#).

For the [graphical user interface \(GUI\)](#) compliant to the [freedesktop.org](#) standards, the `*.desktop` files in the `/usr/share/applications` directory provide necessary attributes for the GUI menu display of each program. Each package which is compliant to [Freedesktop.org](#)'s xdg menu system installs its menu data provided by `"*.desktop"` under `/usr/share/applications/`. Modern desktop environments which are compliant to [Freedesktop.org](#) standard use these data to generate their menu using the `xdg-utils` package. See `/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 3:

- `$XDG_DATA_HOME` é desconfigurada. (É usado o valor predefinido de `$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

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 GNOME3: "Definições" → "Sistema" → "Detalhes" → "Aplicações Predefinidas"
 - Ambiente KDE: "K" → "Centro de Controle" → "Componentes do KDE" → "selecionar Componente"
 - Navegador Iceweasel: "Editar" → "Preferências" → "Aplicações"
 - mc(1): `/etc/mc/mc.ext`
- Variáveis de ambiente como a `$BROWSER`, `$EDITOR`, `$VISUAL` e `$PAGER` (veja `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 X como a sua aplicação preferida, deve criar uma aplicação X 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.

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
```

valor do sinal	nome do sinal	acção	note
0	---	no signal is sent (see <code>kill(2)</code>)	check if process is running
1	<code>SIGHUP</code>	terminate the process	disconnected terminal (signal hang up)
2	<code>SIGINT</code>	terminate the process	interrupt from keyboard (CTRL - C)
3	<code>SIGQUIT</code>	terminate the process and dump core	quit from keyboard (CTRL - \)
9	<code>SIGKILL</code>	terminate the process	unblockable kill signal
15	<code>SIGTERM</code>	terminate the process	blockable termination signal

Tabela 9.11: Lista dos sinais frequentemente usados para o comando kill

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/username"` com o comando `"crontab -e"`.

Aqui está um exemplo de um ficheiro `crontab(5)`.

```
# use /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"`.

[Systemd](#) has low level capability to schedule programs to run without `cron` daemon. For example, `/lib/systemd/system/apt-daily.service` and `/lib/systemd/system/apt-daily.service` set up daily apt download activities. See `systemd.timer(5)`.

9.4.15 Tecla Alt-SysRq

Pressing `Alt-SysRq` (`PrtScr`) followed by one keys does the magic of rescuing control of the system.

tecla que segue a Alt-SysRq	descrição da acção
k	kill all processes on the current virtual console (SAK)
s	sincroniza todos os sistemas de ficheiros montados para evitar corrupção de dados
u	remontar todos os sistemas de ficheiros montados em modo de apenas-leitura (umount)
r	restaurar o teclado a partir de modo raw após crash do X

Tabela 9.12: List of notable SAK command keys

See more on [Linux kernel user’s and administrator’s guide » Linux Magic System Request Key Hacks](#)

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 ssincroniza e umounts (desmonta) todos os sistemas de ficheiros montados.

The current (2021) Debian amd64 Linux kernel has `/proc/sys/kernel/sysrq=438=0b110110110`:

- 2 = 0x2 - enable control of console logging level (ON)
- 4 = 0x4 - enable control of keyboard (SAK, unraw) (ON)
- 8 = 0x8 - enable debugging dumps of processes etc. (OFF)
- 16 = 0x10 - enable sync command (ON)
- 32 = 0x20 - enable remount read-only (ON)
- 64 = 0x40 - enable signaling of processes (term, kill, oom-kill) (OFF)
- 128 = 0x80 - allow reboot/poweroff (ON)
- 256 = 0x100 - allow nicing of all RT tasks (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)` shows who is logged on.
- `w(1)` shows who is logged on and what they are doing.
- `last(1)` shows listing of last logged in user.
- `lastb(1)` mostra a listagem dos últimos utilizadores a falharem o início 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 aparelhos tipo PCI ([AGP](#), [PCI-Express](#), [CardBus](#), [ExpressCard](#), etc.), o `lspci(8)` (provavelmente com a opção `-nn`) é um bom início 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:200, I:991	196	Utilitários PCI do Linux: lspci(8)
usbutils	V:73, I:858	325	Utilitários USB do Linux: lsusb(8)
nvme-cli	V:2, I:10	779	NVMe utilities for Linux: nvme(1)
pcmciautils	V:11, I:17	97	Utilitários PCMCIA par Linux: pccardctl(8)
scsitools	V:0, I:3	375	colecção de ferramentas para gestão de hardware SCSI: lsscsi(8)
procinfo	V:0, I:12	135	informação do sistema obtida de <code>"/proc"</code> : lsdev(8)
lshw	V:11, I:96	850	informação acerca da configuração do hardware: lshw(1)
discover	V:37, I:948	90	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.

pacote	popcon	tamanho	descrição
console-setup	V:133, I:960	419	tipo de letra da consola Linux e utilitários da tabela de teclas
x11-xserver-utils	V:272, I:516	511	Utilitários do servidor X: xset(1) , xmodmap(1)
acpid	V:135, I:281	176	daemon para gerir eventos entregues pelo Advanced Configuration and Power Interface (ACPI)
acpi	V:15, I:266	45	utilitário para mostrar informação em aparelhos ACPI
sleepd	V:0, I:0	86	daemon para pôr um portátil em modo de adormecimento durante a inatividade
hdparm	V:446, I:717	256	optimização do acesso ao disco rígido (veja Secção 9.6.9)
smartmontools	V:150, I:208	2117	controlar e monitorizar sistemas de armazenamento a usar S.M.A.R.T.
setserial	V:5, I:9	117	colecção de ferramentas para gestão de portas série
memtest86+	V:1, I:28	2391	colecção de ferramentas para gestão de hardware de memória
scsitools	V:0, I:3	375	colecção de ferramentas para gestão de hardware SCSI
setcd	V:0, I:1	35	optimização de acesso a drives de discos compactos
big-cursor	I:1	27	cursores de rato maiores para o X

Tabela 9.14: Lista de ferramentas de configuração do hardware

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\(GMT\)](#).

If the hardware time is set to UTC, change the setting to "UTC=yes" in the "/etc/default/rcS".

O seguinte reconfigura a zona horária usada pelo sistema Debian.

```
# dpkg-reconfigure tzdata
```

Se desejar atualizar a hora do sistema através da rede, considere usar o serviço [NTP](#) como pacotes como os `ntp`, `ntpd` 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 Public Services Project](#)
- 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 configuração do terminal

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".

9.5.7 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.

Application softwares may be configured not only to access sound devices directly but also to access them via some standardized sound server system. Currently, PulseAudio, JACK, and PipeWire are used as sound server system. See [Debian wiki page on Sound](#) for the latest situation.

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:337, I:456	2283	utilitários para configurar e utilizar ALSA
oss-compat	V:1, I:27	20	compatibilidade de OSS sob ALSA para prevenir erros de <code>"/dev/dsp não encontrado"</code>
pipewire	V:59, I:78	30	audio and video processing engine multimedia server - metapackage (unify JACK and PulseAudio)
pipewire-bin	V:67, I:76	1848	audio and video processing engine multimedia server - programs (unify JACK and PulseAudio)
pipewire-audio-client-libraries	V:0, I:1	380	audio and video processing engine multimedia server - audio client libraries to replace ALSA, JACK and PulseAudio (experimental)
pulseaudio	V:357, I:453	6774	PulseAudio server (standard)
libpulse0	V:292, I:586	971	PulseAudio client library (standard)
jackd	V:3, I:26	9	servidor (baixa latência) JACK Audio Connection Kit. (JACK)
libjack0	V:1, I:12	338	biblioteca (baixa latência) JACK Audio Connection Kit. (JACK)
libgststreamer1.0-0	V:374, I:567	5283	GStreamer : motor de som do GNOME
libphonon4	I:101	680	Phonon : motor de som do KDE

Tabela 9.15: Lista de pacotes de som

9.5.8 desativar o protector de ecrã (screensaver)

Para desativar o protector de ecrã, utilize os seguintes comandos.

ambiente	comando
A consola do Linux	<code>setterm -powersave off</code>
O X Window (desativar o protector de ecrã)	<code>xset s off</code>
O X Window (desativar o dpms)	<code>xset -dpms</code>
O X Window (GUI de configuração do protector de ecrã)	<code>xscreensaver-command -prefs</code>

Tabela 9.16: Lista de comandos para desativar o protector de ecrã

9.5.9 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
```

9.5.10 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

9.5.11 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](#).
- [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.

pacote	popcon	tamanho	descrição
logcheck	V:8, I:10	102	daemon para enviar as anomalias nos ficheiros de log do sistema ao administrador por mail
debsums	V:5, I:44	108	utilitário para verificar os pacotes instalados contra chaves de verificação MD5
chkrootkit	V:5, I:24	971	detector de rootkit
clamav	V:12, I:57	775	utilitário de anti-vírus para Unix - interface de linha de comandos
tiger	V:2, I:3	7822	relatar vulnerabilidades de segurança do sistema
tripwire	V:2, I:3	11618	verificador de integridade de ficheiros e diretórios
john	V:2, I:12	460	ferramenta activa de crack de palavras-passe
aide	V:1, I:2	2049	Ambiente de Detecção de Intrusão Avançado - binário estático
integrit	V:0, I:0	329	programa de verificação de integridade de ficheiros
crack	V:0, I:1	152	programa de adivinhação de palavra-passe

Tabela 9.18: Lista de ferramentas para segurança do sistema e verificação de integridade

9.6 Dicas de armazenamento de dados

Arrancar o seu sistema com [live CDs](#) de Linux ou [CDs de instalação de debian](#) em modo de recuperação torna fácil para si reconfigurar o armazenamento de dados no seu aparelho de arranque.

You may need to `umount(8)` some devices manually from the command line before operating on them if they are automatically mounted by the GUI desktop system.

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(1x)` 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.

Older PCs use the classic [Master Boot Record \(MBR\)](#) scheme to hold [disk partitioning](#) data in the first sector, i.e., [LBA](#) sector 0 (512 bytes).

Recent PCs with [Unified Extensible Firmware Interface \(UEFI\)](#), including Intel-based Macs, use [GUID Partition Table \(GPT\)](#) scheme to hold [disk partitioning](#) data not in the first sector.

Apesar do `fdisk(8)` ter sido o standard como ferramenta de particionamento de disco, o `parted(8)` está a substituí-lo.



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)`.

pacote	popcon	tamanho	descrição
util-linux	V:899, I:999	4599	vários utilitários de sistema incluindo <code>fdisk(8)</code> e <code>cfdisk(8)</code>
parted	V:367, I:548	307	GNU Parted programa de redimensionamento de partições do disco
gparted	V:16, I:127	2063	Editor de partições do GNOME baseado na <code>libparted</code>
gdisk	V:298, I:501	880	partition editor for the GPT/MBR hybrid disk
kpartx	V:19, I:30	87	programa para criar mapeamentos de aparelho para partições

Tabela 9.19: Lista de pacotes de gestão de partições do disco

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

Although reconfiguration of your partition or activation order of removable storage media may yield different names for partitions, you can access them consistently. This is also helpful if you have multiple disks and your BIOS/UEFI doesn't give them consistent device names.

- `mount(8)` com a opção `"-U"` pode montar um aparelho 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 aparelho especial de bloco com `blkid(8)`.
You can also probe it and other information with `"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 `unde1` 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:616, I:999	1511	utilitários para os sistemas de ficheiros ext2/ext3/ext4
btrfs-progs	V:44, I:68	3959	utilitários para o sistema de ficheiros btrfs
reiserfsprogs	V:15, I:29	1132	utilitários para o sistema de ficheiros Reiserfs
zfsutils-linux	V:19, I:23	1497	utilities for the OpenZFS filesystem
dosfstools	V:132, I:508	315	utilitários para o sistema de ficheiros FAT . (Microsoft: MS-DOS, Windows)
exfatprogs	V:0, I:1	161	utilities for the exFAT filesystem maintained by Samsung.
exfat-fuse	V:25, I:429	71	read/write exFAT filesystem (Microsoft) driver for FUSE.
exfat-utls	V:19, I:430	231	utilities for the exFAT filesystem maintained by the exfat-fuse author.
xfsprogs	V:22, I:101	3281	utilitários para o sistema de ficheiros XFS . (SGI: IRIX)
ntfs-3g	V:227, I:494	1483	read/write NTFS filesystem (Microsoft: Windows NT, ...) driver for FUSE.
jfsutils	V:1, I:12	1577	utilitários para o sistema de ficheiros JFS . (IBM: AIX, OS/2)
reiser4progs	V:0, I:4	1373	utilitários para o sistema de ficheiros Reiser4
hfsprogs	V:0, I:7	389	utilitários para os sistemas de ficheiros HFS e HFS Plus . (Apple: Mac OS)
zerofree	V:3, I:100	25	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 específicas para não o fazer.

[Btrfs](#) status can be found at [Debian wiki on btrfs](#) and [kernel.org wiki on btrfs](#). It is expected to be the next default filesystem after the [ext4](#) filesystem.

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 aparelho de bloco em vez de nomes vulgares de aparelhos de bloco, tal como `"/dev/sda1"`, `"/dev/sda2"`,...

Since Linux 2.6.30, the kernel defaults to the behavior provided by `"relatime"` option.

See `fstab(5)` and `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/hda1"` mostra o conteúdo do super-bloco do sistema de ficheiros em `"/dev/hda1"`.
- A execução de `"sudo tune2fs -c 50 /dev/hda1"` muda a frequência das verificações do sistema de ficheiros (execução do `fsck` durante o arranque) para cada 50 arranques em `"/dev/hda1"`.
- A execução de `"sudo tune2fs -j /dev/hda1"` adiciona capacidade de journal ao sistema de ficheiros, isto é, conversão de `ext2` para `ext3` em `"/dev/hda1"`. (Faça isto no sistema de ficheiros desmontado.)
- A execução de `"sudo tune2fs -O extents,uninit_bg,dir_index /dev/hda1 && fsck -pf /dev/hda1"` converte-o de `ext3` para `ext4` em `"/dev/hda1"`. (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 Otimização 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/hda`, por `hdparm -tT /dev/hda`. Para algum disco rígido ligado com (E)IDE, pode acelerá-lo com `hdparm -q -c3 -d1 -u1 -m16 /dev/hda` 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 Otimização de disco de estado sólido (SSD)

[Solid state drive \(SSD\)](#) is auto detected now.

Reduce unnecessary disk accesses to prevent disk wear out by mounting `"tmpfs"` on volatile data path in `/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/hda`.
4. Verifique o resultado de `smartctl -a /dev/hda` para ver se a funcionalidade [SMART](#) está atualmente ligada.
 - Se não, active-o com `smartctl -s on -a /dev/hda`.
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`.
 - restart the `smartd(8)` daemon by `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 aparelhos 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 aparelho 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 aparelho não montado, ex., a segunda drive SCSI ou serial ATA `"/dev/sdb"`, pode ser feito a usar o `cp(1)` ou o `dd(1)` com o seguinte.

```
# cp /dev/sdb disk.img
# dd if=/dev/sdb of=disk.img
```

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/hda of=mbr.img bs=512 count=1
# dd if=/dev/hda of=mbr-nopart.img bs=446 count=1
# dd if=/dev/hda 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 tem um aparelho SCSI ou serial ATA como disco de arranque, substitua `"/dev/hda"` por `"/dev/sda"`.

Se está a criar uma imagem de uma partição de disco do disco original, substitua `"/dev/hda"` por `"/dev/hda1"` etc.

9.7.2 Escrever directamente no disco

O ficheiro de imagem de disco `"disk.img"` pode ser escrito para um aparelho desmontado, ex. a segunda drive SCSI `"/dev/sdb"` como tamanho correspondente, com o seguinte.

```
# dd if=disk.img of=/dev/sdb
```

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
```

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 [aparelho loop](#) como a seguir.

```
# losetup -v -f partition.img
Loop device is /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 [aparelho loop](#). Como o aparelho loop não gere partições por predefinição, temos que o redefinir como a seguir.

```
# modinfo -p loop # verify kernel capability
max_part:Maximum number of partitions per loop device
max_loop:Maximum number of loop devices
# losetup -a # verify nothing using the loop device
# rmmod loop
# modprobe loop max_part=16
```

Agora, o aparelho loop pode lidar com 16 partições (máximo).

```
# losetup -v -f disk.img
Loop device is /dev/loop0
# fdisk -l /dev/loop0

Disk /dev/loop0: 5368 MB, 5368709120 bytes
255 heads, 63 sectors/track, 652 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x452b6464

    Device Boot      Start         End      Blocks   Id  System
/dev/loop0p1          1           600     4819468+   83  Linux
/dev/loop0p2        601           652       417690    83  Linux
# mkdir -p /mnt/loop0p1
# mount -t ext4 /dev/loop0p1 /mnt/loop0p1
# mkdir -p /mnt/loop0p2
# mount -t ext4 /dev/loop0p2 /mnt/loop0p2
...hack...hack...hack
# umount /dev/loop0p1
# umount /dev/loop0p2
# losetup -d /dev/loop0
```

Alternativamente, podem-se fazer efeitos semelhantes ao usar os aparelhos [device mapper](#) criados pelo kpartx(8) do pacote kpartx como a seguir.

```
# kpartx -a -v disk.img
...
# mkdir -p /mnt/loop0p2
# mount -t ext4 /dev/mapper/loop0p2 /mnt/loop0p2
...
...hack...hack...hack
```

```
# umount /dev/mapper/loop0p2
...
# kpartx -d /mnt/loop0
```

Nota

Também pode montar uma única partição de tal imagem de disco com o [aparelho loop](#) a usar um offset para saltar o [MBR](#) etc. Mas isto é mais inclinado a erros.

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 -f -v disk.img
Loop device is /dev/loop3
# zerofree /dev/loop3
# cp --sparse=always disk.img new.img
```

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
```

Pode criar um sistema de ficheiros ext4 nesta imagem de disco "disk.img" a usar o [aparelho loop](#) como a seguir.

```
# losetup -f -v disk.img
Loop device is /dev/loop1
# mkfs.ext4 /dev/loop1
...hack...hack...hack
# losetup -d /dev/loop1
$ 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 aparelhos criados pelo [aparelho loop](#) ou o [mapeador de aparelhos](#) 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

O ficheiro de imagem [ISO9660](#), "cd.iso", a partir da árvore de directórios fonte em "source_directory" 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-debian` como árvore de directórios em "source_directory" 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 aparelho 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)` disponibilizado pelo [cdrkit](#).

Pode procurar um aparelho 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 aparelho, ex. "/dev/hda", a usar o `wodim(1)` com o seguinte.

```
# wodim -v -eject dev=/dev/hda 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/hda cd.iso
```

Dica

Se o seu ambiente montar CDs automaticamente, desmonte-o com "`sudo umount /dev/hda`" 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:901, I:999	17478	pacote básico que tem <code>od(1)</code> para despejar ficheiros (HEX, ASCII, OCTAL, ...)
bsdmainutils	V:53, I:953	27	pacote utilitário que tem <code>hd(1)</code> para despejar ficheiros (HEX, ASCII, OCTAL, ...)
hexedit	V:1, I:12	73	editor binário e visualizador (HEX, ASCII)
bless	V:0, I:4	1028	editor hexadecimal cheio de funcionalidades (GNOME)
okteta	V:1, I:14	1504	editor hexadecimal cheio de funcionalidades (KDE4)
ncurses-hexedit	V:0, I:2	132	editor binário e visualizador (HEX, ASCII, EBCDIC)
beav	V:0, I:0	133	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.

pacote	popcon	tamanho	descrição
mtools	V:9, I:82	399	utilitários para sistemas de ficheiros MSDOS sem os montar
hfsutils	V:0, I:6	184	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

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
par2	V:4, I:26	271	Parity Archive Volume Set, para verificação e reparação de ficheiros
dvdisaster	V:0, I:2	1741	protecção de dados contra percas/riscos/envelhecimento para medias CD/DVD
dvbackup	V:0, I:0	413	ferramenta de backup que usa câmaras de filmar MiniDV (a disponibilizar rsbep(1))
vdmfec	V:0, I:0	97	recuperar blocos perdidos a usar o Forward Error Correction

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:3, I:37	1426	utilitários para sondagem de partições e recuperação de discos
magicrescue	V:0, I:3	259	utilitário para recuperar ficheiros ao procurar por bytes mágicos
scalpel	V:0, I:4	87	frugal, entalhador de ficheiros de alta performance
myrescue	V:0, I:3	83	recuperar dados de discos rígidos danificados
extundelete	V:0, I:11	147	utilitários para recuperar ficheiros apagados no sistema de ficheiros ext3/4
ext4magic	V:0, I:5	233	utilitários para recuperar ficheiros apagados no sistema de ficheiros ext3/4
ext3grep	V:0, I:3	293	ferramenta para ajudar a recuperar ficheiros apagados no sistema de ficheiros ext3
scrounge-ntfs	V:0, I:3	50	programa de recuperação de dados para sistemas de ficheiros NTFS
gzrt	V:0, I:0	33	conjunto de ferramentas de recuperação gzip
sleuthkit	V:2, I:25	1511	ferramentas para análise forense. (Sleuthkit)
autopsy	V:0, I:1	1027	interface gráfica para o SleuthKit
foremost	V:0, I:7	100	aplicação forense para recuperar dados
guymager	V:0, I:1	1035	ferramenta de imagem forense baseada em Qt
dcfldd	V:0, I:5	106	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 `unde1` 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 crashes 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 de um aparelho inteiro tipo disco rígido, ex., pen de memória USB em `/dev/sda`.

**Cuidado**

Primeiro verifique a localização da sua pen de memória USB com o `mount(8)` antes de executar os comandos aqui. O aparelho 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
```

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/hda"`, `"/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 duma pen 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 pen de memória 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 ligações rígidas. 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 apagados 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](#) facilita a encriptação automática de dados via módulos nativos do kernel Linux com esforços mínimos do utilizador usando [device-mapper](#).

pacote	popcon	tamanho	descrição
cryptsetup	V:41, I:77	407	utilitários para aparelhos de bloco encriptados (dm-crypt / LUKS)
cryptmount	V:4, I:5	228	utilitários para aparelhos de bloco encriptados (dm-crypt / LUKS) com focagem na montagem/desmontagem por utilizadores normais
fscrypt	V:0, I:0	4688	utilities for Linux filesystem encryption (fscrypt)
libpam-fscrypt	V:0, I:0	4904	PAM module for Linux filesystem encryption (fscrypt)

Tabela 9.25: Lista de utilitários de encriptação de dados



Cuidado

Data encryption costs CPU time etc. Encrypted data becomes inaccessible if its password is lost. Please weigh its benefits and costs.

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 aparelhos de massa amovíveis, por exemplo, uma pen USB em `/dev/sdx`, a usar [dm-crypt/LUKS](#). Simplesmente formate-a como a seguir.

```
# 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
```

Then, it can be mounted just like normal one on to `/media/username/disk_label`, except for asking password (see Secção [10.1.7](#)) under modern desktop environment using the `udisks2` package. The difference is that every data written to it is encrypted.

You may alternatively format media in different filesystem, e.g., ext4 with `mkfs.ext4 /dev/mapper/sdx1`. If btrfs is used instead, the `udisks2-btrfs` package needs to be installed. For these filesystems, the file ownership and permissions may need to be configured.

Nota

If you are really paranoid for the security of data, you may initialize device with `badblocks -c 1024 -s -w -t random -v /dev/sdx` command. This operation is very time consuming and wear the life of the storage device.

9.9.2 Partição swap encriptada com dm-crypt

Vamos assumir que o seu `/etc/fstab` original contém o seguinte.

```
/dev/sda7 swap sw 0 0
```

Pode ativar a encriptação da partição swap a usar o [dm-crypt](#) com o seguinte.

```
# aptitude install cryptsetup
# swapoff -a
# echo "cswap /dev/sda7 /dev/urandom swap" >> /etc/crypttab
# perl -i -p -e "s/\/dev\/sda7\/\/dev\/mapper\/cswap/" /etc/fstab
# systemctl restart cryptdisks
...
# swapon -a
```

9.9.3 Montar discos encriptados com dm-crypt/LUKS

Uma partição de disco criada com dm-crypt/LUKS em `"/dev/sdc5"` 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.

If you are reading this documentation, you probably don't need to compile Linux kernel by yourself.

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](#))

See "[The Linux kernel user's and administrator's guide](#) » [The kernel's command-line parameters](#)" for the detail.

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

For compiling some kernel-specific programs such as the kernel modules from the external source and the automounter daemon (`amd`), you must include path to the corresponding kernel headers, e.g. `"-I/usr/src/linux-particular-version/include/"`, to your command line.

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	I:500	20	pacotes essenciais para construir pacotes Debian: make, gcc, ...
bzip2	V:147, I:973	122	utilitários de compressão e descompressão para ficheiros bz2
libncurses5-dev	I:113	6	bibliotecas de programadores e documentos para ncurses
git	V:312, I:494	36515	git: sistema de controle de versão distribuído usado pelo kernel Linux
fakeroot	V:32, I:518	228	disponibiliza um ambiente de falso-root para construção de pacotes como não-root
initramfs-tools	V:442, I:989	112	ferramenta para construir uma initramfs (específico de Debian)
dkms	V:99, I:210	295	suporte de módulos de kernel dinâmicos (DKMS) (genérico)
module-assistant	V:1, I:30	391	helper tool to make module package (Debian specific)
devscripts	V:8, I:54	2654	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)` (or its short form `m-a`) helps users to build and install module package(s) easily for one or more custom kernels.

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 http://www.kernel.org/pub/linux/kernel/v3.11/linux-version.tar.bz2
$ tar -xjvf linux-version.tar.bz2
$ cd linux-version
$ cp /boot/config-version .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_a" e "debian/rules.gen".

```
$ sudo apt-get build-dep linux
$ apt-get source linux
$ cd linux-3.*
$ fakeroot make -f debian/rules.gen binary-arch_i386_none_686
```

Veja mais informação:

- Wiki Debian: [KernelFAQ](#)
- Wiki Debian: [DebianKernel](#)
- Debian Linux Kernel Handbook: <https://kernel-handbook.debian.net>

9.10.5 Controladores de hardware e firmware

The hardware driver is the code running on the main CPUs of the target system. Most hardware drivers are available as free software now and are included in the normal Debian kernel packages in the `main` area.

- 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.)
- controlador de [Softmodem](#)
 - os pacotes `martian-modem` e `sl-modem-dkms` (`non-free`)

The firmware is the code or data loaded on the device attach to the target system (e.g., CPU [microcode](#), rendering code running on GPU, or [FPGA](#) / [CPLD](#) data, ...). Some firmware packages are available as free software but many firmware packages are not available as free software since they contain sourceless binary data. Installing these firmware data is essential for the device to function as expected.

- The firmware data packages containing data loaded to the volatile memory on the target device.
 - `firmware-linux-free` (`main`)
 - `firmware-linux-nonfree` (`non-free`)
 - `firmware-linux-*` (`non-free`)
 - `firmware` (`non-free`)
 - `intel-microcode` (`non-free`)
 - `amd64-microcode` (`non-free`)
- The firmware update program packages which update data on the non-volatile memory on the target device.
 - `fwupd` (`main`): Firmware update daemon which downloads firmware data from [Linux Vendor Firmware Service](#).
 - `gnome-firmware` (`main`): GTK front end for `fwupd`
 - `plasma-discover-backend-fwupd` (`main`): Qt front end for `fwupd`

Por favor note que os pacotes `non-free` e `contrib` não fazem parte do sistema Debian. A configuração de acesso para ativar ou desativar as áreas `non-free` e `contrib` é descrita em Secção [2.1.4](#). Deve estar ciente dos contras associados à utilização de pacotes `non-free` e `contrib`, conforme é descrito em Secção [2.1.5](#).

Please also note that the firmware data downloaded by `fwupd` from [Linux Vendor Firmware Service](#) and loaded to the running Linux kernel may be `non-free`.

9.11 Sistema virtualizado

O uso de sistema virtualizado permite-nos correr várias instâncias do sistema simultaneamente num único hardware.

Dica

Veja <http://wiki.debian.org/SystemVirtualization>.

9.11.1 Virtualization and emulation tools

There are several [virtualization](#) and emulation tool platforms.

- Complete [hardware emulation](#) packages such as ones installed by the [games-emulator](#) metapackage
- Mostly CPU level emulation with some I/O device emulations such as [QEMU](#)
- Mostly CPU level virtualization with some I/O device emulations such as [Kernel-based Virtual Machine \(KVM\)](#)
- OS level container virtualization with the kernel level support such as [LXC \(Linux Containers\)](#), [Docker](#), ...
- OS level filesystem access virtualization with the system library call override on the file path such as [chroot](#)
- OS level filesystem access virtualization with the system library call override on the file ownership such as [fakeroot](#)
- OS API emulation such as [Wine](#)
- Interpreter level virtualization with its executable selection and run-time library overrides such as [virtualenv](#) and [venv](#) for Python

The container virtualization uses Secção 4.7.4 and it is the backend technology of Secção 7.6.

Here are some packages to help you to setup the virtualized system.

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.

- 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 [aparelho loop](#) ou aparelho do [device mapper](#) (veja Secção 9.7.3).
 - Para imagens de disco suportadas pelo [QEMU](#), monte-as como [aparelhos de bloco de rede](#) (veja Secção 9.11.3).
-

pacote	popcon	tamanho	descrição
schroot	V:7, I:9	2708	ferramenta especializada para executar pacotes binários Debian em chroot
sbuid	V:1, I:4	271	ferramenta para construir pacotes binários Debian a partir de fontes Debian
debootstrap	V:5, I:62	298	bootstrap um sistema Debian básico (escrito em sh)
cdebootstrap	V:0, I:2	116	bootstrap um sistema Debian (escrito em C)
virt-manager	V:10, I:43	2298	Virtual Machine Manager : aplicação de ambiente de trabalho para gerir máquinas virtuais
libvirt-clients	V:46, I:64	1130	programas para a biblioteca libvirt
games-emulator	I:0	26	games-emulator : Debian's emulators for games
bochs	V:0, I:1	7194	Bochs : emulador PC IA-32
qemu	I:31	96	QEMU : emulador de processador genérico rápido
qemu-system	I:21	97	QEMU : binários de emulação de sistema completo
qemu-user	V:0, I:11	89683	QEMU : binários de emulação em modo de utilizador
qemu-utils	V:11, I:108	6077	QEMU : utilitários
qemu-kvm	V:8, I:55	107	KVM : virtualização completa em hardware x86 com virtualização assistida por hardware
virtualbox	V:11, I:14	107018	VirtualBox : solução de virtualização x86 em i386 e amd64
xen-tools	V:0, I:4	727	ferramentas para gerir o servidor virtual XEN do debian
wine	V:16, I:76	191	Wine : Windows API Implementation (suite standard)
dosbox	V:2, I:18	2702	DOSBox : emulador x86 com gráficos Tandy/Herc/CGA/EGA/VGA/SVGA, som e DOS
dosemu	V:0, I:2	4891	DOSEMU : O Emulador de DOS do Linux
lxc	V:11, I:14	18971	Ferramentas de utilizador para Linux containers
python3-venv	V:2, I:50	6	venv for creating virtual python environments (system library)
python3-virtualenv	V:9, I:64	410	virtualenv for creating isolated virtual python environments
python3-pipx	I:0	NOT_FOUND	pipx for installing python applications in isolated environments

Tabela 9.27: Lista de ferramentas de virtualização

- 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 [aparelho 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`: O `QEMU` suporta os seguintes formatos de discos: `raw`, `qcow2`, `qcow`, `vmdk`, `vdi`, `bochs`, `cow` (modo-de-utilizador de Linux de copiar-ao-escrever), `parallels`, `dmg`, `cloop`, `vpc`, `vvfat` (VFAT virtual) e `aparelho_máquina`.

O [aparelho de bloco em rede](#) pode suportar partições do mesmo modo que o [aparelho 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

If you wish to try a new Debian environment from a terminal console, I recommend you to use [chroot](#). This enables you to run console applications of Debian `unstable` and `testing` without usual risks associated and without rebooting. `chroot(8)` is the most basic way.



Cuidado

Examples below assumes both parent system and chroot system share the same amd64 CPU architecture.

Although you can manually create a `chroot(8)` environment using `debootstrap(1)`. But this requires non-trivial efforts.

The `sbuild` package to build Debian packages from source uses the chroot environment managed by the `schroot` package. It comes with helper script `sbuild-createchroot(1)`. Let's learn how it works by running it under `script(1)` as follows.

```
$ sudo mkdir -p /srv/chroot
$ sudo sbuild-createchroot -v --include=eatmydata,ccache unstable /srv/chroot/unstable- ↵
amd64-sbuild http://deb.debian.org/debian
```

You see how `debootstrap(8)` populates system data for `unstable` environment under `"/srv/chroot/unstable-amd64-sbuild"` for a minimal build system.

You can login to this environment using `schroot(1)`.

```
$ sudo schroot -v -c chroot:unstable-amd64-sbuild
```

You see how a system shell running under `unstable` environment is created.

Nota

The `"/usr/sbin/policy-rc.d"` file which always exits with 101 prevents daemon programs to be started automatically on the Debian system. See `"/usr/share/doc/sysv-rc/README.policy-rc.d.gz"`.

Nota

Some programs under `chroot` may require access to more files from the parent system to function than `sbuild-createchroot` provides as above. For example, `"/sys"`, `"/etc/passwd"`, `"/etc/group"`, `"/var/run/utmp"`, `"/var/log/wtmp"`, etc. may need to be bind-mounted or copied.

Dica

The `sbuild` package helps to construct a `chroot` system and builds a package inside the `chroot` using `schroot` as its backend. It is an ideal system to check build-dependencies. See more on [sbuild at Debian wiki](#) and [sbuild configuration example in "Guide for Debian Maintainers"](#).

9.11.5 Sistemas de vários ambientes de trabalho

If you wish to try a new GUI Desktop environment of any OS, I recommend you to use [QEMU](#), [KVM](#), or [VirtualBox](#) on a Debian `stable` system to run multiple desktop systems safely using [virtualization](#). These enable you to run any desktop applications including ones of Debian `unstable` and `testing` without usual risks associated with them and without rebooting. The configuration of these tools are relatively straight forward.

Since pure [QEMU](#) is very slow, it is recommended to accelerate it with [KVM](#) when the host system supports it.

A imagem de disco virtual `"virtdisk.qcow2"` que contém um sistema Debian para o [QEMU](#) pode ser criada a usar o [instalador de debian em pequenos CDs](#) como a seguir.

```
$ wget http://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
...
```

Veja mais dicas em [Debian wiki: QEMU](#).

O [VirtualBox](#) vem com ferramentas GUI [Qt](#) e é bastante intuitivo. As suas ferramentas GUI e de linha de comandos estão explicadas em [Manual do Utilizador do VirtualBox](#) e [Manual do Utilizador do VirtualBox \(PDF\)](#).

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.

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 aparelhos 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.

GNU tar(1):

```
$ tar -cvJf archive.tar.xz ./source
$ tar -xvJf archive.tar.xz
```

Alternativamente, pelo seguinte.

```
$ find ./source -xdev -print0 | tar -cvJf archive.tar.xz --null -F -
```

cpio(1):

```
$ find ./source -xdev -print0 | cpio -ov --null > archive.cpio; xz archive.cpio
$ zcat archive.cpio.xz | cpio -i
```


pacote	popcon	tamanho	extensão	comando	comentário
tar	V:914, I:999	3152	.tar	tar(1)	o arquivador standard (de facto standard)
cpio	V:489, I:998	1144	.cpio	cpio(1)	arquivador estilo Unix System V, usar com o find(1)
binutils	V:164, I:673	97	.ar	ar(1)	arquivador para a criação de bibliotecas estáticas
fastjar	V:2, I:25	183	.jar	fastjar(1)	arquivador para Java (estilo zip)
pax	V:12, I:24	170	.pax	pax(1)	novos arquivadores standard do POSIX, um compromisso entre tar e cpio
gzip	V:891, I:999	242	.gz	gzip(1) , zcat(1) , ...	LZ77 utilitário de compressão do GNU (o standard de facto)
bzip2	V:147, I:973	122	.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:2, I:27	149	.lzma	lzma(1)	LZMA compression utility with higher compression ratio than gzip(1) (deprecated)
xz-utils	V:453, I:980	612	.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:4, I:24	1902	.zstd	zstd(1) , zstdcat(1) , ...	Zstandard fast lossless compression utility
p7zip	V:79, I:454	987	.7z	7zr(1) , p7zip(1)	7-Zip arquivador de arquivos com alta taxa de compressão (compressão LZMA)
p7zip-full	V:102, I:469	4664	.7z	7z(1) , 7za(1)	7-Zip arquivador de ficheiros com rácio de compressão alto (LZMA compressão e outros)
lzop	V:10, I:85	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:49, I:427	623	.zip	zip(1)	InfoZIP : ferramenta de compressão e arquivo do DOS
unzip	V:132, I:792	385	.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:901, I:999	17478	GNU cp	copia localmente ficheiros e diretórios ("a" para ser recursivo)
openssh-client	V:809, I:997	4298	scp	copia remotamente ficheiros e diretórios (cliente, "r" para ser recursivo)
openssh-server	V:701, I:844	1567	sshd	copia ficheiros e diretórios remotamente (servidor remoto)
rsync	V:311, I:564	677		sincronização remota a salvaguarda de 1 via
unison	V:3, I:16	14		sincronização remota a salvaguarda de 2 vias

Tabela 10.2: Lista de ferramentas de cópia e sincronização

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.domínio`

`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
```

GNU `tar(1)`:

```
# (cd ./source && tar cf - . ) | (cd /dest && tar xvpf - )
# (cd ./source && tar cf - . ) | ssh user@host.dom '(cd /dest && tar xvpf - )'
```

`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/times

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 sobreescrever. 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.

10.1.7 Aparelho de armazenamento amovível

Aparelhos 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](#)
- [PC Card](#)

Os ambientes de trabalho modernos tais como o GNOME e KDE podem montar automaticamente estes aparelhos amovíveis sem uma entrada correspondente no `/etc/fstab`.

- `udisks2` package provides a daemon and associated utilities to mount and unmount these devices.
- [D-bus](#) cria eventos para iniciar processos automáticos.
- [PolicyKit](#) disponibiliza os privilégios necessários.

Dica

Os aparelhos auto-montados podem ter a opção de montagem `uhelp=` que é utilizada por `umount(8)`.

Dica

A auto-montagem em ambientes de trabalho modernos apenas acontece quando esses aparelhos amovíveis não estão listados em `/etc/fstab`.

Mount point under modern desktop environment is chosen as `/media/username/disk_label` which can be customized by the following.

- `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 aparelho dele gerado dinamicamente tal como `/dev/sdc`. Se desejar manter o nó de aparelho dele, desmonte-o com o comando `umount(8)` na linha de comandos da shell.

filesystem name	typical usage scenario
FAT12	partilha de dados em várias plataformas em disquetes (<32MiB)
FAT16	partilha de dados em várias plataformas em aparelhos como pequenos discos rígidos (<2GiB)
FAT32	partilha de dados em várias plataformas em aparelhos como grandes discos rígidos (<8TiB, suportado por mais recente que MS Windows95 OSR2)
exFAT	cross platform sharing of data on the large hard disk like device (<512TiB, supported by WindowsXP, Mac OS X Snow Leopard 10.6.5, and Linux kernel since 5.4 release)
NTFS	partilha de dados em várias plataformas em aparelhos 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 aparelhos tipo disco rígido com sistemas Linux mais antigos
ext3	partilha de dados em aparelhos tipo disco rígido com sistemas Linux mais antigos
ext4	partilha de dados em aparelhos de tipo disco rígido com sistemas Linux atuais
btrfs	sharing of data on the hard disk like device with current Linux systems with read-only snapshots

Tabela 10.3: Lista de hipóteses de sistemas de ficheiros para aparelhos de armazenamento amovíveis com cenários de utilização típica

10.1.8 Escolha de sistema de ficheiros para partilhar dados

Quando partilha dados com outros sistemas via aparelhos 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.

Dica

Veja Secção [9.9.1](#) para partilha de dados em várias plataformas a usar encriptação ao nível do aparelho.

O sistema de ficheiros FAT é suportado pela maioria dos sistemas operativos modernos e é bastante útil para objetivos de trocas de dados via aparelhos tipo disco rígido.

Quando formatar aparelhos 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.
 - Apenas o nome de aparelho dele, ex. `"/dev/sda1"` para FAT16
 - A opção explícita e o nome de aparelho 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 <code>smb.conf(5)</code> e O HOWTO Oficial do Samba 3.x.x e Guia de Referência ou o pacote <code>samba-doc</code>
NFS sistema de ficheiros montado em rede com o kernel do Linux	partilhar ficheiros via "Rede Unix/Linux", veja <code>exports(5)</code> 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 segura 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.

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/"`
 - Softwares 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
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.
 - O valor dos dados quando perdidos
 - Recursos necessários para o backup: humano, hardware, software, ...
 - Modo de falha e a possibilidade dele

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 exactamente o que está a fazer, eles são enormes quantidades de dados desnecessários.

Para o armazenamento seguro de dados, os dados devem estar pelo menos em partições de disco diferentes de preferência em discos e máquinas diferentes para sobreviverem à corrupção do sistema de ficheiros. Os dados importantes ficam melhor armazenados em medias onde só se escreve uma vez, como os CD/DVD-R para prevenir serem sobrescritos por acidente. (veja Secção 9.8 para como escrever na media de armazenamento a partir da linha de comandos shell. O ambiente gráfico de trabalho GNOME dá-lhe acesso fácil via menu: "Acessórios → Criador de CD/DVD".)

Nota

Pode desejar parar alguns daemons de aplicação como o MTA (veja Secção 6.2.4) enquanto faz cópias de segurança (backups) dos dados.

Nota

Deve ter cuidados extra com o backup e restauro de ficheiros de dados relacionados com identidade como os `/etc/ssh/ssh_host_dsa_key`, `/etc/ssh/ssh_host_rsa_key`, `~/.gnupg/*`, `~/.ssh/*`, `/etc/passwd`, `/etc/shadow`, `/etc/fetchmailrc`, `popularity-contest.conf`, `/etc/ppp/pap-secrets` e `/etc/exim4/passwd.client`. Alguns destes dados não podem ser regenerados ao inserir a mesma cadeia de entrada ao sistema.

Nota

Se executa uma tarefa cron como um processo de utilizador, tem de restaurar os ficheiros no diretório `/var/spool/cron/crontabs` e reiniciar o `cron(8)`. Veja Secção 9.4.14 para `cron(8)` e `crontab(1)`.

10.2.1 Suites de utilitários de backup

Aqui está uma lista selecionada de suites de utilitários de backup notáveis disponíveis no sistema Debian.

pacote	popcon	tamanho	descrição
dump	V:1, I:6	352	4.4 BSD dump(8) e restore(8) para sistemas de ficheiros ext2/ext3/ext4
xfsdump	V:0, I:9	854	dump e restore com xfsdump(8) e xfsrestore(8) para sistema de ficheiros XFS em GNU/Linux e IRIX
backupninja	V:3, I:4	367	sistema de meta-backup leve e extensível
bacula-common	V:10, I:14	2158	Bacula : salvaguarda, recuperação e verificação em rede - ficheiros de suporte comum
bacula-client	I:3	183	Bacula : salvaguarda, recuperação e verificação em rede - meta-pacote cliente
bacula-console	V:1, I:4	107	Bacula : salvaguarda, recuperação e verificação em rede - consola de texto
bacula-server	I:1	183	Bacula : salvaguarda, recuperação e verificação em rede - meta-pacote servidor
amanda-common	V:1, I:2	10030	Amanda : Advanced Maryland Automatic Network Disk Archiver (Bibliotecas)
amanda-client	V:1, I:2	1088	Amanda : Advanced Maryland Automatic Network Disk Archiver (Cliente)
amanda-server	V:0, I:0	1075	Amanda : Advanced Maryland Automatic Network Disk Archiver (Servidor)
backup-manager	V:1, I:1	571	ferramenta de salvaguarda de linha de comandos
backup2l	V:0, I:1	114	ferramenta de baixa manutenção para salvaguarda/restauro para medias montáveis (baseado em disco)
backuppc	V:3, I:3	3183	BackupPC é um sistema de grau empresarial de alta performance para fazer salvaguardas a PCs (baseado em disco)
duplicity	V:9, I:19	1834	salvaguarda incremental (remoto)
flexbackup	V:0, I:0	243	salvaguarda incremental (remoto)
rdiff-backup	V:6, I:14	733	salvaguarda incremental (remoto)
restic	V:1, I:4	22540	salvaguarda incremental (remoto)
slbackup	V:0, I:0	151	salvaguarda incremental (remoto)

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.
- Regular backups of user data can be realized by a simple script (Secção [10.2.2](#)).

Ferramentas básicas descritas em Secção [10.1.1](#) e Secção [10.1.2](#) podem ser usadas facilitar o backup do sistema via scripts personalizados. Tal script pode ser melhorado com o seguinte.

- O pacote `restic` permite salvaguardas incrementais (remotas).
- O pacote `rdiff-backup` permite salvaguardas incrementais (remotas).
- O pacote `dump` ajuda a arquivar e restaurar o sistema de ficheiros completo de modo incremental e eficiente.

Dica

Veja os ficheiros em `/usr/share/doc/dump/` e ["está o dump mesmo obsoleto?"](#) para aprender acerca do pacote `dump`.

10.2.2 Personal backup

For a personal Debian desktop system running `testing` suite, I only need to protect personal and critical data. I reinstall system once a year anyway. Thus I see no reason to backup the whole system or to install a full featured backup utility.

At the same time, it is very valuable to have frequent recent snapshots of personal data and system configuration, and occasional full backups of personal data.

I usually make these snapshots and backups with a simple shell script [bss](#). This script is a short shell which uses standard utilities: `btrfs subvolume snapshot`, `rsync`, `cp`.

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"`.

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.

See Secção [9.9](#) on [dm-crypt](#) and [ecryptfs](#) which implement automatic data encryption infrastructure via Linux kernel modules.

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 envia a minha chave `"1DD8D791"` para o popular servidor de chaves `"hkp://keys.gnupg.net"`.

```
$ gpg --keyserver hkp://keys.gnupg.net --send-keys 1DD8D791
```

Um bom servidor de chaves predefinido configurado em `"~/ .gnupg/gpg.conf"` (ou na antiga localização `"~/ .gnupg/options"`) contém o seguinte.

```
keyserver hkp://keys.gnupg.net
```

pacote	popcon	tamanho	comando	descrição
gnupg	V:498, I:942	864	gpg(1)	GNU Privacy Guard - ferramenta de encriptação e assinatura OpenPGP
gpgv	V:890, I:999	882	gpgv(1)	GNU Privacy Guard - ferramenta de verificação de assinaturas
paperkey	V:1, I:13	58	paperkey(1)	extrai apenas a informação secreta de chaves secretas OpenPGP
cryptsetup	V:41, I:77	407	cryptsetup(8) ...	utilities for dm-crypt block device encryption supporting LUKS
coreutils	V:901, I:999	17478	md5sum(1)	computa e verifica o resumo da mensagem MD5
coreutils	V:901, I:999	17478	sha1sum(1)	computa e verifica o resumo da mensagem SHA1
openssl	V:814, I:994	1465	openssl(1ssl)	computa o resumo da mensagem com "openssl dgst" (OpenSSL)

Tabela 10.6: Lista de ferramentas de infraestrutura da segurança de dados

comando	descrição
gpg --gen-key	gerar uma chave nova
gpg --gen-revoke meu_ID_utilizador	gera chave de revogação para meu_ID_utilizador
gpg --edit-key ID_utilizador	edita chave interativamente, "help" para ajuda
gpg -o ficheiro --export	exporta todas as chaves para ficheiro
gpg --import ficheiro	importa todas as chaves de ficheiro
gpg --send-keys ID_utilizador	envia chave de ID_utilizador para servidor de chaves
gpg --recv-keys ID_utilizador	recupera chave de ID_utilizador do servidor de chaves
gpg --list-keys ID_utilizador	lista chaves de ID_utilizador
gpg --list-sigs ID_utilizador	lista assinaturas de ID_utilizador
gpg --check-sigs ID_utilizador	verifica assinaturas de ID_utilizador
gpg --fingerprint ID_utilizador	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

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-bu`.

10.3.2 Usa GnuPG em ficheiros

Aqui estão exemplos para usar comandos do [GNU Privacy Guard](#) em ficheiros.

comando	descrição
<code>gpg -a -s ficheiro</code>	assina ficheiro em ficheiro.asc blindado de ASCII
<code>gpg --armor --sign ficheiro</code>	, ,
<code>gpg --clearsign ficheiro</code>	mensagem com assinatura clara
<code>gpg --clearsign file mail foo@example.org</code>	envia por mail uma mensagem com assinatura clara para foo@example.org
<code>gpg --clearsign --not-dash-escaped patchfile</code>	ficheiro patch com assinatura clara
<code>gpg --verify ficheiro</code>	verifica ficheiro com assinatura clara
<code>gpg -o ficheiro.sig -b ficheiro</code>	cria assinatura separada
<code>gpg -o ficheiro.sig --detach-sig ficheiro</code>	, ,
<code>gpg --verify ficheiro.sig ficheiro</code>	verifica ficheiro com file.sig
<code>gpg -o crypt_file.gpg -r nome -e ficheiro</code>	encriptação de chave pública destinada a nome a partir de ficheiro para crypt_file.gpg binário
<code>gpg -o crypt_file.gpg --recipient nome --encrypt ficheiro</code>	, ,
<code>gpg -o crypt_file.asc -a -r nome -e ficheiro</code>	encriptação de chave pública destinada a nome a partir de ficheiro para crypt_file.asc blindado de ASCII
<code>gpg -o crypt_file.gpg -c ficheiro</code>	encriptação simétrica a partir de ficheiro para crypt_file.gpg
<code>gpg -o crypt_file.gpg --symmetric ficheiro</code>	, ,
<code>gpg -o crypt_file.asc -a -c ficheiro</code>	encriptação simétrica destinada a nome a partir de ficheiro para crypt_file.asc blindado de ASCII
<code>gpg -o ficheiro -d crypt_file.gpg -r nome</code>	desencriptação
<code>gpg -o ficheiro --decrypt crypt_file.gpg</code>	, ,

Tabela 10.9: Lista de 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

The gnupg plugin let you run GnuPG transparently for files with extension ".gpg", ".asc", and ".ppg".¹

```
$ sudo aptitude install vim-scripts
$ echo "packadd! gnupg" >> ~/.vim/vimrc
```

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.

10.4 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
```

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
```

¹If you use "~/.vimrc" instead of "~/.vim/vimrc", please substitute accordingly.

pacote	popcon	tamanho	comando	descrição
patch	V:123, I:721	248	<code>patch(1)</code>	aplica ficheiro diff a um original
vim	V:102, I:404	3286	<code>vimdiff(1)</code>	compara dois ficheiros lado a lado no vim
imediff	V:0, I:0	170	<code>imediff(1)</code>	ferramenta de fusão de 2 ou 3 vias interactiva de écran completo
meld	V:14, I:38	3065	<code>meld(1)</code>	compara e funde ficheiros (GTK)
wigggle	V:0, I:0	174	<code>wigggle(1)</code>	aplica patches rejeitadas
diffutils	V:883, I:993	1598	<code>diff(1)</code>	compara ficheiros linha a linha
diffutils	V:883, I:993	1598	<code>diff3(1)</code>	compara e junta três ficheiros linha a linha
quilt	V:3, I:32	788	<code>quilt(1)</code>	gere séries de patches
wdiff	V:8, I:69	644	<code>wdiff(1)</code>	mostra diferenças de palavras entre ficheiros de texto
diffstat	V:14, I:146	81	<code>diffstat(1)</code>	produz um histograma de alterações feitas pelo diff
patchutils	V:16, I:143	232	<code>combinediff(1)</code>	cria uma patch cumulativa de duas patches incrementais
patchutils	V:16, I:143	232	<code>dehtmldiff(1)</code>	extrai um diff de uma página HTML
patchutils	V:16, I:143	232	<code>filterdiff(1)</code>	extrai ou executa diffs de um ficheiro diff
patchutils	V:16, I:143	232	<code>fixcvsdiff(1)</code>	corrige ficheiros diff criados pelo CVS que o <code>patch(1)</code> interpreta mal
patchutils	V:16, I:143	232	<code>flipdiff(1)</code>	troca a ordem de duas patches
patchutils	V:16, I:143	232	<code>grepdiff(1)</code>	mostra que ficheiros são modificados por uma patch que corresponde a um regex
patchutils	V:16, I:143	232	<code>interdiff(1)</code>	mostra as diferenças entre dois ficheiros diff unificados
patchutils	V:16, I:143	232	<code>lsdiff(1)</code>	mostra quais ficheiros são modificados por uma patch
patchutils	V:16, I:143	232	<code>recountdiff(1)</code>	recalcula contagens e offsets em diffs de contexto unificado
patchutils	V:16, I:143	232	<code>rediff(1)</code>	corrige os offsets e as contagens de um diff editado manualmente
patchutils	V:16, I:143	232	<code>splitdiff(1)</code>	separa patches incrementais
patchutils	V:16, I:143	232	<code>unwrapdiff(1)</code>	desembaralha patches cujas linhas foram alteradas para arrumação de palavras
dirdiff	V:0, I:2	166	<code>dirdiff(1)</code>	mostra diferenças e funde alterações entre árvores de diretórios
docdiff	V:0, I:0	555	<code>docdiff(1)</code>	compara dois ficheiros palavra a palavra / caractere a caractere
makepatch	V:0, I:0	102	<code>makepatch(1)</code>	gera ficheiros de patch extensos
makepatch	V:0, I:0	102	<code>applypatch(1)</code>	aplica ficheiros de patch extensos

Tabela 10.10: Lista de ferramentas de fusão de código fonte

10.4.3 Interactive merge

If you have two versions of a source code, you can perform 2-way merge interactively using `imediff(1)` by the following.

```
$ imediff -o file.merged file.old file.new
```

If you have three versions of a source code, you can perform 3-way merge interactively using `imediff(1)` by the following.

```
$ imediff -o file.merged file.yours file.base file.theirs
```

10.5 Git

Git is the tool of choice these days for the [version control system \(VCS\)](#) since Git can do everything for both local and remote source code management.

O Debian disponibiliza serviços Git livres via [Serviço Debian Salsa](#).. A sua documentação encontra-se em <https://wiki.debian.org/Salsa>.

Here are some Git related packages.

pacote	popcon	tamanho	comando	descrição
git	V:312, I:494	36515	git(7)	Git, o sistema de controlo de revisão distribuído, rápido e escalável
gitk	V:5, I:41	1749	gitk(1)	explorador GUI de repositórios Git com histórico
git-gui	V:1, I:23	2343	git-gui(1)	GUI para Git (Nenhum histórico)
git-email	V:0, I:11	993	git-send-email(1)	envia uma colecção de patches como email a partir do Git
git-buildpackage	V:1, I:11	4187	git-buildpackage(1)	o empacotamento Debian com o Git
dgit	V:0, I:1	496	dgit(1)	git interoperability with the Debian archive
imediff	V:0, I:0	170	git-ime(1)	interactive git commit split helper tool
stgit	V:0, I:0	603	stg(1)	quilt no topo do git (Python)
git-doc	I:14	12063	N/D	documentação oficial para o Git
gitmagic	I:1	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

10.5.1 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
```

You may also customize the Git default behavior by the following.

```
$ 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
```

10.5.2 Basic Git commands

Git operation involves several data.

- The working tree which holds user facing files and you make changes to them.
 - The changes to be recorded must be explicitly selected and staged to the index. This is `git add` and `git rm` commands.
- The index which holds staged files.
 - Staged files will be committed to the local repository upon the subsequent request. This is `git commit` command.
- The local repository which holds committed files.
 - Git records the linked history of the committed data and organizes them as branches in the repository.
 - The local repository can send data to the remote repository by `git push` command.
 - The local repository can receive data from the remote repository by `git fetch` and `git pull` commands.
 - * The `git pull` command performs `git merge` or `git rebase` command after `git fetch` command.
 - * Here, `git merge` combines two separate branches of history at the end to a point. (This is default of `git pull` without customization and may be good for upstream people who publish branch to many people.)
 - * Here, `git rebase` creates one single branch of sequential history of the remote branch one followed by the local branch one. (This is `pull.rebase true` customization case and may be good for rest of us.
- The remote repository which holds committed files.
 - The communication to the remote repository uses secure communication protocols such as SSH or HTTPS.

The working tree is files outside of the `.git/` directory. Files inside of the `.git/` directory hold the index, the local repository data, and some git configuration text files.

Here is an overview of main Git commands.

10.5.3 Git tips

Here are some Git tips.



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

If a local branch which has been pushed to remote repository is rebased or squashed, pushing this branch has risks and requires `--force` option. This is usually not an acceptable for `main` branch but may be acceptable for a topic branch before merging to `main` branch.

Git command	função
<code>git init</code>	cria o repositório (local)
<code>git clone URL</code>	clone the remote repository to a local repository with the working tree
<code>git pull origin main</code>	update the local <code>main</code> branch by the remote repository <code>origin</code>
<code>git add .</code>	add file(s) in the working tree to the index for pre-existing files in index only
<code>git add -A .</code>	add file(s) in the working tree to the index for all files including removals
<code>git rm filename</code>	remove file(s) from the working tree and the index
<code>git commit</code>	commit staged changes in the index to the local repository
<code>git commit -a</code>	add all changes in the working tree to the index and commit them to the local repository (add + commit)
<code>git push -u origin branch_name</code>	update the remote repository <code>origin</code> by the local <code>branch_name</code> branch (initial invocation)
<code>git push origin branch_name</code>	update the remote repository <code>origin</code> by the local <code>branch_name</code> branch (subsequent invocation)
<code>git diff treeish1 treeish2</code>	show difference between <i>treeish1</i> commit and <i>treeish2</i> commit
<code>gitk</code>	GUI display of VCS repository branch history tree

Tabela 10.12: Main Git commands

**Cuidado**

Invocar um sub-comando `git` directamente como "`git-xyz`" a partir da linha de comandos foi descontinuado desde o início de 2006.

Dica

If there is a executable file `git-foo` in the path specified by `$PATH`, entering "`git foo`" without hyphen to the command line invokes this `git-foo`. This is a feature of the `git` command.

10.5.4 Referências do Git

Veja o seguinte.

- [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`)
- [Everyday GIT With 20 Commands Or So](#) (`/usr/share/doc/git-doc/giteveryday.html`)
- [Magia do Git](#) (`/usr/share/doc/gitmagic/html/index.html`)

10.5.5 Other version control systems

The [version control systems \(VCS\)](#) is sometimes known as the revision control system (RCS), or the software configuration management (SCM).

Here is a summary of the notable other non-Git VCS on the Debian system.

Git command line	função
<code>gitk --all</code>	see complete Git history and operate on them such as resetting HEAD to another commit, cheery-picking patches, creating tags and branches ...
<code>git stash</code>	get the clean working tree without losing data
<code>git remote -v</code>	check settings for remote
<code>git branch -vv</code>	check settings for branch
<code>git status</code>	show working tree status
<code>git config -l</code>	list git settings
<code>git reset --hard HEAD; git clean -x -d -f</code>	revert all working tree changes and clean them up completely
<code>git rm --cached filename</code>	revert staged index changed by <code>git add filename</code>
<code>git reflog</code>	get reference log (useful for recovering commits from the removed branch)
<code>git branch new_branch_name HEAD@{6}</code>	create a new branch from reflog information
<code>git remote add new_remote URL</code>	add a <code>new_remote</code> remote repository pointed by URL
<code>git remote rename origin upstream</code>	rename the remote repository name from <code>origin</code> to <code>upstream</code>
<code>git branch -u upstream/branch_name</code>	set the remote tracking to the remote repository <code>upstream</code> and its branch name <code>branch_name</code> .
<code>git remote set-url origin https://foo/bar.git</code>	change URL of <code>origin</code>
<code>git remote set-url --push upstream DISABLED</code>	disable push to <code>upstream</code> (Edit <code>.git/config</code> to re-enable)
<code>git checkout -b topic_branch ; git push -u topic_branch origin</code>	make a new <code>topic_branch</code> and push it to <code>origin</code>
<code>git branch -m oldname newname</code>	rename local branch name
<code>git push -d origin branch_to_be_removed</code>	remove remote branch (new method)
<code>git push origin :branch_to_be_removed</code>	remove remote branch (old method)
<code>git checkout --orphan unconnected</code>	create a new <code>unconnected</code> branch
<code>git rebase -i origin/main</code>	reorder/drop/squish commits from <code>origin/main</code> to clean branch history
<code>git reset HEAD^; git commit --amend</code>	squash last 2 commits into one
<code>git checkout topic_branch ; git merge --squash topic_branch</code>	squash entire <code>topic_branch</code> into a commit
<code>git fetch --unshallow --update-head-ok origin '+refs/heads/*:refs/heads/*'</code>	convert a shallow clone to the full clone of all branches
<code>git ime</code>	split the last commit into a series of file-by-file smaller commits etc. (<code>imediff</code> package required)
<code>git repack -a -d; git prune</code>	repack the local repository into single pack (this may limit chance of lost data recovery from erased branch etc.)

Tabela 10.13: Git tips

pacote	popcon	tamanho	ferramenta	Tipo VCS	comentário
mercurial	V:6, I:44	1053	Mercurial	distribuído	DVCS em Python e algum C
darcs	V:0, I:7	23159	Darcs	distribuído	DVCS com álgebra inteligente de patches (lento)
bazaar	V:1, I:15	28	Bazaar	distribuído	DVCS influenced by tla written in Python (historic)
tla	V:0, I:2	1011	GNU arch	distribuído	DVCS mainly by Tom Lord (historic)
subversion	V:18, I:104	4873	Subversion	remoto	"CVS done right", newer standard remote VCS (historic)
cvs	V:4, I:38	4620	CVS	remoto	previous standard remote VCS (historic)
tkcvs	V:0, I:1	1498	CVS, ...	remoto	ecrã GUI de árvores de repositório VCS (CVS, Subversion, RCS)
rcs	V:3, I:19	562	RCS	local	" Unix SCCS done right" (historic)
cssc	V:0, I:2	2044	CSSC	local	clone of the Unix SCCS (historic)

Tabela 10.14: List of other version control system tools

Capítulo 11

Conversão de dados

São descritas ferramentas e dicas para converter formatos de dados no sistema Debian.

As ferramentas baseadas em standards são muitas 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:946, I:999	12772	conjunto e caracteres (charset)	converter codificação de texto entre locais por <code>iconv(1)</code> (fundamental)
recode	V:3, I:25	603	charset+eol	conversor de codificação de texto entre locais (versátil, com mais nomes alternativos (alias) e funcionalidades)
konwert	V:1, I:52	134	conjunto e caracteres (charset)	conversor de codificação de texto entre locais (imaginativo)
nkf	V:0, I:11	358	conjunto e caracteres (charset)	tradutor de conjunto de caracteres para Japonês
tcs	V:0, I:0	518	conjunto e caracteres (charset)	tradutor de conjunto de caracteres
unaccent	V:0, I:0	29	conjunto e caracteres (charset)	substitui letras acentuadas pelo seu equivalente não acentuado
tofrodos	V:1, I:23	55	eol	conversor de formato de texto entre DOS e Unix: <code>de dos(1)</code> e <code>para dos(1)</code>
macutils	V:0, I:0	298	eol	conversor de formato de texto entre Macintosh e Unix: <code>de mac(1)</code> e <code>para mac(1)</code>

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

Some encodings are only supported for the data conversion and are not used as locale values (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
ENCN=iso-8859-1
for x in *;
do
  mv "$x" "$(echo "$x" | iconv -f $ENCN -t utf-8)"
done
```

A variável "`$ENCN`" 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`".

plataforma	código EOL	controle	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

11.1.4 conversão EOL

O formato de ficheiro de texto, especificamente o código de fim de linha (EOL), é dependente da plataforma.

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	<code>bsdmainutils</code>	<code>coreutils</code>
expande tab para espaços	<code>"col -x"</code>	<code>expand</code>
contrai tab a partir de espaços	<code>"col -h"</code>	<code>unexpand</code>

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 http://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:25, I:252	2367	html → texto	Conversor de HTML para texto com o comando "w3m -dump"
html2text	V:2, I:29	274	html → texto	Conversor de HTML para texto avançado (ISO 8859-1)
lynx	V:11, I:97	1948	html → texto	Conversor de HTML para texto com o comando "lynx -dump"
elinks	V:5, I:27	1767	html → texto	Conversor de HTML para texto com o comando "elinks -dump"
links	V:5, I:38	2250	html → texto	Conversor de HTML para texto com o comando "links -dump"
links2	V:1, I:14	5418	html → texto	Conversor de HTML para texto com o comando "links2 -dump"
catdoc	V:22, I:125	675	MSWord → texto, tex	converte ficheiros do MSWord para texto simples ou ps
antiword	V:1, I:10	589	MSWord → texto, ps	converte ficheiros do MSWord para texto simples ou ps
pstopdf	V:0, I:2	126	ps/pdf → texto	extraí texto de ficheiros PostScript e PDF
unhtml	V:0, I:0	43	html → texto	remove as etiquetas de marcas de um ficheiro HTML
odt2txt	V:1, I:7	60	odt → texto	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:18, I:439	32134	destaque	MACRO do Vim para converter código fonte em HTML com ":source \$VIMRUNTIME/syntax/html.vim"
cxref	V:0, I:0	1193	c → html	conversor de programa C para latex e HTML (linguagem C)
src2tex	V:0, I:0	622	destaque	converte muitos códigos fonte para TeX (linguagem C)
source-highlight	V:0, I:7	1992	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:1, I:10	1083	destaque	converte muitos códigos fonte para HTML, XHTML, RTF, LaTeX, TeX ou ficheiros XSL-FO com destaques (C++)
grc	V:0, I:3	208	texto → cor	colorizador genérico para tudo (Python)
python-docutils	V:9, I:112	1771	texto → qualquer	Formatador de documento de Texto Re-Estruturado para XML (Python)
pandoc	V:8, I:48	151714	texto → qualquer	conversor geral de markup (Haskell)
markdown	V:0, I:8	57	texto → html	formatador de documentos de texto markdown para (X)HTML (Perl)
txt2html	V:0, I:3	259	texto → html	conversor de texto para HTML (Perl)
asciidoc	I:12	81	texto → qualquer	formatador de documentos de texto AsciiDoc para XML/HTML (Python)

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
&	& : ampersand

Tabela 11.7: Lista de entidades predefinidas para XML



Cuidado

"<" ou "&" não podem ser usados em atributos ou elementos.

Nota

When SGML style user defined entities, e.g. "&some-tag:", are used, the first definition wins over others. The entity definition is expressed in "*!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:14, I:264	2133	xml	definição de tipo de documento XML (DTD) para DocBook
docbook-xsl	V:12, I:154	14870	xml/xslt	folhas de estilo XSL para processar XML do DocBook para vários formatos de saída com XSLT
xsltproc	V:14, I:100	160	xslt	processador de linha de comandos XSLT (XML → XML, HTML, texto simples, etc.)
xmlto	V:1, I:22	130	xml/xslt	conversor de XML-para-qualquer com XSLT
fop	V:1, I:20	291	xml/xsl-fo	converter ficheiros Docbook XML para PDF
dblatex	V:2, I:15	4643	xml/xslt	converte ficheiros do Docbook para documentos DVI, PostScript, PDF com o XSLT
dbtoepub	V:0, I:0	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:2, I:36	1019	dsssl	ISO/IEC 10179:1996 processador DSSSL standard (mais recente)
docbook-dsssl	V:1, I:21	2604	xml/dsssl	folhas de estilo DSSSL para processar XML do DocBook para vários formatos de saída com DSSSL
docbook-utils	V:0, I:15	281	xml/dsssl	utilitários para ficheiros do DocBook incluindo a conversão para outros formatos (HTML, RTF, PS, man, PDF) com comandos <code>docbook2*</code> com DSSSL
sgml2x	V:0, I:0	90	SGML/dsssl	conversor de SGML e XML que usa folhas de estilo DSSSL

Tabela 11.9: Lista de ferramentas DSSSL

Dica

O `ye lp` do [GNOME](#) por vezes é jeitoso para ler ficheiros XML do [DocBook](#) directamente pois ele renderiza directamente 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, I:2	138	manpage → html	conversor de manual (manpage) para HTML (suporte a CGI)
doclifter	V:0, I:0	451	troff → xml	conversor de troff para DocBook XML
texi2html	V:0, I:7	1833	texi → html	conversor de Texinfo para HTML
info2www	V:1, I:3	76	info → html	conversor de info do GNU para HTML (suporte a CGI)
wv	V:0, I:6	717	MSWord → qualquer	conversor de documentos Microsoft Word para HTML, LaTeX, etc.
unrtf	V:0, I:4	148	rtf → html	conversor de documentos de RTF para HTML, etc
ooo2dbk	V:0, I:0	217	sxw → xml	conversor de documentos SXW do OpenOffice.org para documentos XML do DocBook
wp2x	V:0, I:0	202	WordPerfect → qualquer	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 The XML data lint

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.

Syntax of XML files and goodness of URLs found in them may be checked.

pacote	popcon	tamanho	função	descrição
libxml2-utils	V:21, I:234	181	xml ↔ html ↔ xhtml	ferramenta de XML de linha de comandos com xml Lint(1) (verificação de sintaxe, reformatação, lint, ...)
tidy	V:1, I:13	84	xml ↔ html ↔ xhtml	Verificador e reformatador de sintaxe HTML
weblint-perl	V:0, I:1	32	lint	Verificado de sintaxe e estilo mínimo para HTML
linklint	V:0, I:0	344	link check	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.

pacote	popcon	tamanho	palavra chave	descrição
texlive	V:3, I:48	72	(La)TeX	sistema TeX para formatação de texto, pre-visualização e impressão
groff	V:3, I:59	11870	troff	O sistema de formato de texto troff do GNU

Tabela 11.12: Lista de ferramentas de formatação de texto

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](#). Pode ler ou imprimir um bom tutorial e referência à [macro](#) `"-me"` em `"/usr/share/doc/groff/"` ao instalar 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 [OOVEOQT](#) 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](#)
- [Guia do Utilizador Local para o teTeX/LaTeX](#)

Quando os documentos ficam maiores, por vezes o TeX pode causar erros. tem de aumentar o tamanho do pool em `"/etc/texmf/texm"` (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 <http://tug.ctan.org/tex-archive/systems/knuth/dist/tex/texbook.tex>. Este ficheiro contém a maioria das macros necessárias. Ouvi que pode processar este documento com o `tex(1)` após comentar as linhas 7 a 10 e a adicionar "`\input manmac \proofmodefalse`". É fortemente recomendado comprar este livro (e todos os outros livros de Donald E. Knuth) em vez de usar a versão online mas a fonte é um grande exemplo de entrada em 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, I:12	191	SGML → manpage	conversor de DocBook SGML para macros roff man
help2man	V:0, I:10	491	texto → manpage	geração automática de manual a partir do <code>--help</code>
info2man	V:0, I:0	135	info → manpage	conversor de info do GNU para POD ou páginas man
txt2man	V:0, I:1	114	texto → 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.

11.4.1 Ghostscript

O núcleo da manipulação de dados imprimíveis é o interpretador `PostScript (PS) Ghostscript` o qual gera imagem em rasterização.

pacote	popcon	tamanho	descrição
ghostscript	V:256, I:576	231	O interpretador de PostScript/PDF Ghostscript GPL
ghostscript-x	V:12, I:63	223	Interpretador de PostScript/PDF Ghostscript GPL - suporte a ecrã X
libpoppler102	V:55, I:89	4274	Biblioteca de renderização de PDF que é um fork do visualizador de PDF xpdf
libpoppler-glib8	V:228, I:463	445	Biblioteca de renderização de PDF (biblioteca de partilha baseada em GLib)
poppler-data	V:102, I:621	13090	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 coligadas de um ficheiro a usar um dos seguintes comandos.

```
$ lp -n 3 -o Collate=True filename
```

```
$ 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=e", "-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.

Dica

The [Internet Message Access Protocol](#) version 4 (IMAP4) server may be used to move mails out from proprietary mail systems if the mail client software can be configured to use IMAP4 server too.

pacote	popcon	tamanho	palavra chave	descrição
poppler-utils	V:245, I:413	689	pdf → ps,text, ...	Utilitários de PDF: pdftops , pdftinfo , pdfimages , pdftotext , pdffonts
psutils	V:6, I:98	219	ps → ps	Ferramentas de conversão de documentos PostScript
poster	V:0, I:4	58	ps → ps	criar grandes posters de páginas PostScript
enscript	V:1, I:19	2132	text → ps, html, rtf	converter texto ASCII para PostScript, HTML, RTF ou Pretty-Print
a2ps	V:0, I:14	3648	text → ps	Conversor de 'Qualquer coisa para PostScript' e pretty-printer
pdftk	I:50	28	pdf → pdf	Ferramenta de conversão de documentos PDF: pdftk
html2ps	V:0, I:3	249	html → ps	conversor de HTML para PostScript
gnuhtml2latex	V:0, I:1	27	html → latex	conversor de html para latex
latex2rtf	V:0, I:6	480	latex → rtf	conversor de documentos LaTeX para RTF que podem ser lidos pelo MS Word
ps2eps	V:3, I:62	99	ps → eps	conversor de PostScript para EPS (PostScript Encapsulado)
e2ps	V:0, I:0	109	text → ps	Conversor de texto para PostScript com suporte a codificação Japonesa
impose+	V:0, I:0	119	ps → ps	Utilitários do PostScript
trueprint	V:0, I:0	146	text → ps	impressão bonita de muitos códigos fonte (C, C++, Java, Pascal, Perl, Pike, Sh e Verilog) para PostScript. (linguagem C)
pdf2svg	V:0, I:4	30	ps → svg	conversor de PDF para formato Gráficos vectoriais escaláveis
pdfstoipe	V:0, I:0	71	ps → ipe	conversor de PDF para formato XML de IPE

Tabela 11.15: Lista de utilitários de dados imprimíveis

pacote	popcon	tamanho	palavra chave	descrição
sharutils	V:4, I:52	1421	mail	shar (1), unshar (1), uuencode (1), uudecode (1)
mpack	V:1, I:17	106	MIME	codificação e decodificação de mensagens MIME : mpack (1) e munpack (1)
tnef	V:1, I:10	110	ms-tnef	descompactar anexos MIME do tipo "application/ms-tnef" o qual é um formato apenas da Microsoft
uudeview	V:0, I:4	109	mail	codificador e decodificador dos seguintes formatos: uuencode , xxencode , BASE64 , quoted printable e BinHex

Tabela 11.16: Lista de pacotes para ajudar na conversão de dados de mail

11.5.1 Noções básicas de dados de mail

Mail ([SMTP](#)) data should be limited to series of 7 bit data. So binary data and 8 bit text data are encoded into 7 bit format with the [Multipurpose Internet Mail Extensions \(MIME\)](#) and the selection of the charset (see 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 texto 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

If your non-Unix mail data is accessible by a non-Debian client software which can talk to the IMAP4 server, you may be able to move them out by running your own IMAP4 server.

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

Os seguintes pacotes para conversão de dados gráficos, edição e ferramentas de organização chamaram a minha atenção.

Dica

Procure mais ferramentas de imagem com a expressão regular `"~Gworks-with::image"` no `aptitude(8)` (veja Secção 2.2.6).

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 de Lempel-Ziv-Welch \(LZW\)](#) expirou. Os utilitários de [Graphics Interchange Format \(GIF\)](#) que usam o método de compressão LZW estão agora livremente disponíveis 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.

pacote	popcon	tamanho	palavra chave	descrição
gimp	V:58, I:315	19706	imagem(bitmap)	GNU Image Manipulation Program
imagemagick	I:373	220	imagem(bitmap)	programas de manipulação de imagens
graphicsmagick	V:2, I:17	5242	imagem(bitmap)	programas de manipulação de imagens (fork do imagemagick)
xsane	V:14, I:165	2346	imagem(bitmap)	Frontend X11 baseado em GTK para o SANE (Scanner Access Now Easy)
netpbm	V:29, I:382	5056	imagem(bitmap)	ferramentas de conversão de gráficos
icoutils	V:18, I:115	221	png ↔ ico(bitmap)	converte ícones e cursores do MS Windows de e para formatos PNG (favicon.ico)
scribus	V:1, I:22	30370	ps/pdf/SVG/...	editor de DTP do Scribus
libreoffice-draw	V:150, I:416	14596	imagem(vector)	suite de escritório do LibreOffice - desenho
inkscape	V:42, I:188	84892	imagem(vector)	editor de SVG (Scalable Vector Graphics)
dia	V:4, I:30	3727	imagem(vector)	editor de diagramas (Gtk)
xfig	V:2, I:14	1793	imagem(vector)	Habilidade para Geração interactiva de figuras sob X11
pstoedit	V:3, I:87	988	ps/pdf → imagem(vector)	conversor de ficheiro PostScript e PDF para gráficos vectoriais editáveis (SVG)
libwmf-bin	V:8, I:191	113	Windows/imagem(vector)	ferramentas de conversão de meta-ficheiros do Windows (dados de gráficos vectoriais)
fig2sxd	V:0, I:0	149	fig → sxd(vector)	converte ficheiros XFig ao formato do OpenOffice.org Draw
unpaper	V:2, I:19	460	imagem → imagem	ferramenta de pós-processamento para páginas digitalizadas em scanner para OCR
tesseract-ocr	V:8, I:37	1507	imagem → texto	software livre de OCR baseado no motor de OCR comercial da HP
tesseract-ocr-eng	V:6, I:37	4032	imagem → texto	Dados de motor OCR: ficheiros de linguagem tesseract-ocr para texto Inglês
gocr	V:1, I:11	531	imagem → texto	software de OCR livre
ocrad	V:0, I:4	303	imagem → texto	software de OCR livre
eog	V:63, I:253	10189	imagem(Exif)	Programa de visualização de gráficos Olho do GNOME
gthumb	V:5, I:21	5483	imagem(Exif)	visualizador e navegador de imagens (GNOME)
geeqie	V:6, I:19	14643	imagem(Exif)	Visualizador de imagens que usa GTK
shotwell	V:16, I:217	6451	imagem(Exif)	organizador de fotos digitais (GNOME)
gtkam	V:0, I:5	1154	imagem(Exif)	Aplicação para recolher media de câmaras digitais (GTK)
gphoto2	V:0, I:12	955	imagem(Exif)	cliente de linha de comandos de camera digital gphoto2
gwenview	V:27, I:94	10651	imagem(Exif)	visualizador de imagens (KDE)
kamera	I:93	798	imagem(Exif)	Suporte para cameras digitais para aplicações do KDE
digikam	V:2, I:12	2646	imagem(Exif)	aplicação de gestão de fotos digitais para KDE
exiv2	V:2, I:40	321	imagem(Exif)	ferramenta de manipulação de meta-dados EXIF/IPTC
exiftran	V:1, I:19	70	imagem(Exif)	transformar imagens jpeg de câmaras digitais
jhead	V:1, I:10	110	imagem(Exif)	manipula a parte de não-imagem de ficheiros JPEG compatíveis com Exif (fotos de câmaras digitais)
exif	V:1, I:12	339	imagem(Exif)	utilitário de linha de comandos para mostrar informação EXIF nos ficheiros JPEG
exiftags	V:0, I:4	292	imagem(Exif)	utilitário para ler etiquetas Exif de ficheiros JPEG de câmaras digitais
exifprobe	V:0, I:4	499	imagem(Exif)	ler meta-dados de imagens digitais
dcraw	V:2, I:18	535	imagem(Raw)	processa e modifica imagens cruas de câmaras digitais
findimagedupes	V:0, I:1	79	image → fingerprint	encontra imagens visualmente semelhantes ou duplicadas
ale	V:0, I:0	839	imagem → imagem	junta imagens para aumentar a fidelidade ou criar mosaicos
imageindex	V:0, I:2	145	imagem(Exif) → html	gera galerias HTML estáticas a partir de imagens
outguess	V:0, I:2	261	jpeg,png	ferramenta de Esteganografia universal
librecad	V:3, I:18	7893	DXF	editor de dados CAD (KDE)
blender	V:3, I:37	79166	blend, TIFF, VRML, ...	editor de conteúdos 3D para animação e etc
mm3d	V:0, I:0	3778	ms3d, obj, dxf, ...	editor de modelos 3D baseado em OpenGL

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).

pacote	popcon	tamanho	palavra chave	descrição
alien	V:2, I:31	163	rpm/tgz → deb	conversor para pacotes alienígenas num pacote Debian
freepwing	V:0, I:0	421	EB → EPWING	conversor de "Electric Book" (popular no Japão) para um formato JIS X 4081 único (um subconjunto de EPWING V1)
calibre	V:8, I:35	55048	qualquer → EPUB	Conversor de e-books e gestor de biblioteca

Tabela 11.18: Lista de ferramentas de conversão de dados variados

Também pode extrair dados do formato RPM com o seguinte.

```
$ rpm2cpio file.src.rpm | cpio --extract
```

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 às 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.

Please consider to use version control system tools. See 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" (<http://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

Many system scripts may be interpreted by any one of [POSIX](#) shells (see Tabela 1.13).

- The default non-interactive POSIX shell `/bin/sh` is a symlink pointing to `/usr/bin/dash` and used by many system programs.
- The default interactive POSIX shell is `/usr/bin/bash`.

Evite escrever um script de shell com **bashisms** ou **zshisms** para fazê-lo portátil entre todas as shells do POSIX. Pode verificar isto a usar o `checkbashisms(1)`.

Bom: POSIX	Evitar: 'bashism'
<code>if ["\$foo" = "\$bar"] ; then ...</code>	<code>if ["\$foo" == "\$bar"] ; then ...</code>
<code>diff -u file.c.orig file.c</code>	<code>diff -u file.c{.orig,}</code>
<code>mkdir /foobar /foobaz</code>	<code>mkdir /foo{bar,baz}</code>
<code>funcname() { ...}</code>	<code>function funcname() { ...}</code>
formato octal: <code>"\377"</code>	formato hexadecimal: <code>"\xff"</code>

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.

parâmetro da shell	valor
<code>\$0</code>	nome da shell ou script de shell
<code>\$1</code>	primeiro(1) argumento shell
<code>\$9</code>	nono(9) argumento shell
<code>\$#</code>	quantidade de parâmetros de posição
<code>"\$*"</code>	<code>"\$1 \$2 \$3 \$4 ..."</code>
<code>"\$@"</code>	<code>"\$1" "\$2" "\$3" "\$4" ...</code>
<code>\$?</code>	estado de saída do comando mais recente
<code>\$\$</code>	PID deste script shell
<code>\$_</code>	PID da tarefa de fundo iniciada mais recentemente

Tabela 12.2: Lista de parâmetros da 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.

formato da expressão do parâmetro	valor se var estiver definido	valor se var não estiver definido
<code>\${var:-cadeia}</code>	<code>"\$var"</code>	<code>"cadeia"</code>
<code>\${var:+cadeia}</code>	<code>"cadeia"</code>	<code>"null"</code>
<code>\${var:=cadeia}</code>	<code>"\$var"</code>	<code>"cadeia"</code> (e corra <code>"var=cadeia"</code>)
<code>\${var:?cadeia}</code>	<code>"\$var"</code>	echo <code>"cadeia"</code> para stderr (e termina com erro)

Tabela 12.3: Lista de expansões de parâmetros de shell

- **com** `":"` = teste de operador para **existe e não nulo**
- **sem** `":"` = teste de operador para apenas **existe**

formato de substituição de parâmetro	resultado
<code>\${var%suffix}</code>	remover o modelo de sufixo menor
<code>\${var%%suffix}</code>	remover o modelo de sufixo maior
<code>\${var#prefix}</code>	remover o modelo de prefixo menor
<code>\${var##prefix}</code>	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"`.

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
<code>-e ficheiro</code>	<i>ficheiro</i> existe
<code>-d ficheiro</code>	<i>ficheiro</i> existe e é um diretório
<code>-f ficheiro</code>	<i>ficheiro</i> existe e é um ficheiro normal
<code>-w ficheiro</code>	<i>ficheiro</i> existe e pode-se escrever nele
<code>-x ficheiro</code>	<i>ficheiro</i> existe e é executável
<code>ficheiro1 -nt ficheiro2</code>	<i>ficheiro1</i> é mais recente que <i>ficheiro2</i> (modificação)
<code>ficheiro1 -ot ficheiro2</code>	<i>ficheiro1</i> é mais antigo que <i>ficheiro2</i> (modificação)
<code>ficheiro1 -ef ficheiro2</code>	<i>ficheiro1</i> e <i>ficheiro2</i> estão no mesmo aparelho e no mesmo número de inode

Tabela 12.5: Lista de operadores de comparação de ficheiros na expressão condicional

equação	condição para retornar o verdadeiro lógico
<code>-z str</code>	o comprimento de <i>str</i> é zero
<code>-n str</code>	o comprimento de <i>str</i> não é zero
<code>str1 = str2</code>	<i>str1</i> and <i>str2</i> são iguais
<code>str1 != str2</code>	<i>str1</i> and <i>str2</i> não são iguais
<code>str1 < str2</code>	<i>str1</i> ordena antes de <i>str2</i> (dependente do locale)
<code>str1 > str2</code>	<i>str1</i> ordena após <i>str2</i> (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 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*
- Metacharacters: `| ; & ()`
- 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 home 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.

12.1.6 Programas utilitários para script de shell

De modo a tornar o seu programa de shell o mais portátil 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

Although `moreutils` may not exist outside of Debian, it offers interesting small programs. Most notable one is `sponge(8)` which is quite useful when you wish to overwrite original file.

See Secção [1.6](#) for examples.

pacote	popcon	tamanho	descrição
dash	V:913, I:993	221	small and fast POSIX-compliant shell for <code>sh</code>
coreutils	V:901, I:999	17478	utilitários de núcleo GNU
grep	V:837, I:999	1091	GNU <code>grep</code> , <code>egrep</code> and <code>fgrep</code>
sed	V:834, I:999	883	GNU <code>sed</code>
mawk	V:392, I:997	242	small and fast <code>awk</code>
debianutils	V:932, I:999	230	utilitários variados específicos do Debian
bsdutils	V:708, I:999	394	utilitários básicos do 4.4BSD-Lite
bsdextrautils	V:80, I:103	384	extra utilities from 4.4BSD-Lite
moreutils	V:12, I:37	237	utilitários Unix adicionais

Tabela 12.7: Lista de pacotes que contém programas utilitários pequenos para scripts de shell

pacote	popcon	tamanho	documentação
dash	V:913, I:993	221	sh : small and fast POSIX-compliant shell for <code>sh</code>
bash	V:796, I:999	6470	sh : "info <code>bash</code> " provided by <code>bash-doc</code>
mawk	V:392, I:997	242	AWK : small and fast <code>awk</code>
gawk	V:359, I:440	2456	AWK : "info <code>gawk</code> " provided by <code>gawk-doc</code>
perl	V:609, I:992	705	Perl : <code>perl(1)</code> and html pages provided by <code>perl-doc</code> and <code>perl-doc-html</code>
libterm-readline-gnu-perl	V:2, I:31	379	Perl extension for the GNU ReadLine/History Library: <code>perlsh(1)</code>
libreply-perl	V:0, I:0	170	REPL for Perl: <code>reply(1)</code>
libdevel-repl-perl	V:0, I:0	237	REPL for Perl: <code>re.pl(1)</code>
python3	V:676, I:901	89	Python : <code>python3(1)</code> and html pages provided by <code>python3-doc</code>
tcl	V:28, I:390	22	Tcl : <code>tcl(3)</code> and detail manual pages provided by <code>tcl-doc</code>
tk	V:28, I:382	22	Tk : <code>tk(3)</code> and detail manual pages provided by <code>tk-doc</code>
ruby	V:141, I:308	35	Ruby : <code>ruby(1)</code> , <code>erb(1)</code> , <code>irb(1)</code> , <code>rdoc(1)</code> , <code>ri(1)</code>

Tabela 12.8: List of interpreter related packages

12.2 Scripting in interpreted languages

When you wish to automate a task on Debian, you should script it with an interpreted language first. The guide line for the choice of the interpreted language is:

- Use `dash`, if the task is a simple one which combines CLI programs with a shell program.
- Use `python3`, if the task isn't a simple one and you are writing it from scratch.
- Use `perl`, `tcl`, `ruby`, ... if there is an existing code using one of these languages on Debian which needs to be touched up to do the task.

If the resulting code is too slow, you can rewrite only the critical portion for the execution speed in a compiled language and call it from the interpreted language.

12.2.1 Debugging interpreted language codes

Most interpreters offer basic syntax check and code tracing functionalities.

- “`dash -n script.sh`” - Syntax check of a Shell script
- “`dash -x script.sh`” - Trace a Shell script
- “`python -m py_compile script.py`” - Syntax check of a Python script
- “`python -mtrace --trace script.py`” - Trace a Python script
- “`perl -I ../libpath -c script.pl`” - Syntax check of a Perl script
- “`perl -d:Trace script.pl`” - Trace a Perl script

For testing code for `dash`, try Secção 9.1.4 which accommodates `bash`-like interactive environment.

For testing code for `perl`, try REPL environment for Perl which accommodates Python-like `REPL (=READ + EVAL + PRINT + LOOP)` environment for Perl.

12.2.2 GUI program with the shell script

The shell script can be improved to create an attractive GUI program. The trick is to use one of so-called dialog programs instead of dull interaction using `echo` and `read` commands.

pacote	popcon	tamanho	descrição
x11-utils	V:171, I:581	712	<code>xmessage(1)</code> : mostra uma mensagem ou questão numa janela (X)
whiptail	V:116, I:996	71	mostra caixas de diálogo amigas do utilizador a partir de scripts de shell (<code>newt</code>)
dialog	V:15, I:118	1222	mostra caixas de diálogo amigas do utilizador a partir de scripts de shell (<code>ncurses</code>)
zenity	V:79, I:393	384	display graphical dialog boxes from shell scripts (GTK)
ssft	V:0, I:0	75	Ferramenta Frontend de Scripts de Shell (wrapper para o <code>zenity</code> , <code>kdialog</code> e <code>dialog</code> com o <code>gettext</code>)
gettext	V:47, I:302	5843	”/usr/bin/gettext.sh”: traduz mensagem

Tabela 12.9: List of dialog programs

Here is an example of GUI program to demonstrate how easy it is just with a shell script.

This script uses `zenity` to select a file (default `/etc/motd`) and display it.

GUI launcher for this script can be created following 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
```

This kind of approach to GUI program with the shell script is useful only for simple choice cases. If you are to write any program with complexities, please consider writing it on more capable platform.

12.2.3 A loucura dos scripts curtos de Perl

In order to process data, `sh` needs to spawn sub-process running `cut`, `grep`, `sed`, etc., and is slow. On the other hand, `perl` has internal capabilities to process data, and is fast. So many system maintenance scripts on Debian use `perl`.

Let's think following one-liner AWK script snippet and its equivalents in 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.
- Perl execution tricks via command line options: `perlrun(1)`
- Perl special variables: `perlvar(1)`

This flexibility is the strength of Perl. At the same time, this allows us to create cryptic and tangled codes. So be careful.

For more crazy Perl scripts, [Perl Golf](#) may be interesting.

12.3 Coding in compiled languages

Here, Seção [12.3.3](#) and Seção [12.3.4](#) are included to indicate how compiler-like program can be written in C language by compiling higher level description into C language.

pacote	popcon	tamanho	descrição
gcc	V:185, I:598	45	GNU C compiler
libc6-dev	V:241, I:615	14358	GNU C Library: Development Libraries and Header Files
g++	V:57, I:519	15	GNU C++ compiler
libstdc++-10-dev	V:16, I:76	17752	GNU Standard C++ Library v3 (development files)
cpp	V:318, I:755	42	GNU C preprocessor
gettext	V:47, I:302	5843	GNU Internationalization utilities
glade	V:0, I:7	1730	GTK User Interface Builder
valac	V:0, I:7	684	C# like language for the GObject system
flex	V:8, I:91	1279	LEX-compatible fast lexical analyzer generator
bison	V:9, I:101	2858	YACC-compatible parser generator
susv2	I:0	16	buscar " The Single UNIX Specifications v2 "
susv3	I:0	16	buscar " The Single UNIX Specifications v3 "
golang	I:21	11	Go programming language compiler
rustc	V:1, I:12	5720	Rust systems programming language
haskell-platform	I:5	12	Standard Haskell libraries and tools
gfortran	V:10, I:97	16	GNU Fortran 95 compiler
fpc	I:3	122	Free Pascal

Tabela 12.10: List of compiler related packages

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"`
- `cada-nome_de_função_da_biblioteca_C(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`.

Tem de fornecer o seu próprio `main()` e `yywrap()`. Caso contrário, o seu programa flex deverá ficar como isto para compilar sem uma biblioteca. Isto é porque o `yywrap` é uma macro e a `%option main` liga implicitamente `%option noyywrap`.

```
%option main
%%
.|\\n      ECHO ;
%%
```

Alternativamente, pode compilar com a opção `-lfl` do linker no final da sua linha de comandos `cc(1)` (como AT&T-Lex com `-ll`). Nenhuma `%opção` é necessária neste caso.

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:9, I:101	2858	gerador de análise GNU LALR
byacc	V:0, I:5	160	Gerador de análise Berkeley LALR
btyacc	V:0, I:0	243	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.

```
%%
%%
```

12.4 Ferramentas de análise de código estático

[Lint](#) like tools can help automatic [static code analysis](#).

[Indent](#) like tools can help human code reviews by reformatting source codes consistently.

[Ctags](#) like tools can help human code reviews by generating an index (or tag) file of names found in source codes.

Dica

Configuring your favorite editor (emacs or vim) to use asynchronous lint engine plugins helps your code writing. These plugins are getting very powerful by taking advantage of [Language Server Protocol](#). Since they are moving fast, using their upstream code instead of Debian package may be a good option.

pacote	popcon	tamanho	descrição
vim-ale	I:0	2022	Asynchronous Lint Engine for Vim 8 and NeoVim
vim-syntastic	I:3	1240	Syntax checking hacks for vim
elpa-flycheck	V:0, I:1	792	modern on-the-fly syntax checking for Emacs
elpa-relint	V:0, I:0	133	Emacs Lisp regexp mistake finder
cppcheck-gui	V:0, I:1	5529	tool for static C/C++ code analysis (GUI)
shellcheck	V:2, I:10	17977	lint tool for shell scripts
pyflakes3	V:1, I:15	23	passive checker of Python 3 programs
pylint	V:2, I:14	1430	Verificador estático de código Python
perl	V:609, I:992	705	interpretador com verificador de código estático interno: B: :Lint(3perl)
rubocop	V:0, I:0	2390	Ruby static code analyzer
clang-tidy	V:0, I:5	22	clang-based C++ linter tool
splint	V:0, I:3	2320	ferramenta para verificação estática de programas C por bugs
flawfinder	V:0, I:0	181	ferramenta para examinar código fonte C/C++ e procurar por fraquezas na segurança
black	V:1, I:3	475	uncompromising Python code formatter
perltidy	V:0, I:5	1674	Perl script indenter and reformatter
indent	V:0, I:11	425	C language source code formatting program
astyle	V:0, I:4	761	Source code indenter for C, C++, Objective-C, C#, and Java
bcpp	V:0, I:1	111	C(++) beautifier
xmllindent	V:0, I:1	53	XML stream reformatter
global	V:0, I:3	1896	Source code search and browse tools
exuberant-ctags	V:4, I:34	349	build tag file indexes of source code definitions

Tabela 12.12: Lista de ferramentas para análise de código estático

12.5 Depuração

Debug is important part of programming activities. Knowing how to debug programs makes you a good Debian user who can produce meaningful bug reports.

pacote	popcon	tamanho	documentação
gdb	V:12, I:113	9799	"info gdb" disponibilizado por gdb-doc
ddd	V:0, I:10	4092	"info ddd" disponibilizado por ddd-doc

Tabela 12.13: List of debug packages

12.5.1 Execução gdb básica

O [depurador](#) principal em Debian é o `gdb(1)` que lhe permite inspecionar 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
```

Good tutorial of `gdb` can be found:

- “[info gdb](#)”
- “[Debugging with GDB](#)” in `/usr/share/doc/gdb-doc/html/gdb/index.html`
- “[tutorial on the web](#)”

Here is a simple example of using `gdb(1)` on a “program” compiled with the “-g” option to produce debugging information.

```
$ 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

Since all installed binaries should be stripped on the Debian system by default, most debugging symbols are removed in the normal package. In order to debug Debian packages with `gdb(1)`, `*-dbgsym` packages need to be installed (e.g. `coreutils-dbgsym` in the case of `coreutils`). The source packages generate `*-dbgsym` packages automatically along with normal binary packages and those debug packages are placed separately in [debian-debug](#) archive. Please refer to [articles on Debian Wiki](#) for more information.

If a package to be debugged does not provide its `*-dbgsym` package, you need to install it after rebuilding it by the following.

```
$ 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 um crash num programa, é uma boa ideia enviar um relatório de bug com informação de backtrace copiada-e-colada.

The backtrace can be obtained by `gdb(1)` using one of the following approaches:

- Crash-in-GDB approach:
 - Run the program from GDB.
 - Crash the program.
 - Type "bt" at the GDB prompt.
- Crash-first approach:
 - Update the `"/etc/security/limits.conf"` file to include the following:

```
* soft core unlimited
```
 - Type `"ulimit -c unlimited"` to the shell prompt.
 - Run the program from this shell prompt.
 - Crash the program to produce a [core dump](#) file.
 - Load the [core dump](#) file to GDB as `"gdb gdb ./program_binary core"`.
 - Type "bt" at the GDB prompt.

For infinite loop or frozen keyboard situation, you can force to crash the program by pressing `Ctrl-\` or `Ctrl-C` or executing `"kill -ABRT PID"`. (See 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 `"$MALLOCCHECK_"` para um valor de 2 (`malloc(3)`). Pode fazer isto enquanto corre o `gdb` ao fazer o seguinte.

```
$ MALLOCCHECK_=2 gdb hello
```

12.5.4 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.

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	obtem 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

```
$ ldd /bin/ls
      librt.so.1 => /lib/librt.so.1 (0x4001e000)
      libc.so.6 => /lib/libc.so.6 (0x40030000)
      libpthread.so.0 => /lib/libpthread.so.0 (0x40153000)
      /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Para 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 Dynamic call tracing tools

There are several dynamic call tracing tools available in Debian. See 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:241, I:615	14358	<code>mtrace(1)</code> : funcionalidades de depuração do malloc em glibc
valgrind	V:5, I:45	80378	depurador e perfilador de memória
electric-fence	V:0, I:5	70	o depurador <code>malloc(3)</code>
libdmalloc5	V:0, I:3	393	biblioteca de depuração de alocação de memória
duma	V:0, I:0	259	library to detect buffer overruns and under-runs in C and C++ programs
leaktracer	V:0, I:2	57	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 Build tools

pacote	popcon	tamanho	documentação
make	V:174, I:603	1592	"info make" disponibilizado por <code>make-doc</code>
autoconf	V:42, I:280	1846	"info autoconf" disponibilizado por <code>autoconf-doc</code>
automake	V:41, I:278	1830	"info automake" disponibilizado por <code>automake1.10-doc</code>
libtool	V:35, I:264	1198	"info libtool" provided by <code>libtool-doc</code>
cmake	V:16, I:122	22944	<code>cmake(1)</code> cross-platform, open-source make system
ninja-build	V:4, I:30	347	<code>ninja(1)</code> small build system closest in spirit to Make
meson	V:2, I:18	2761	<code>meson(1)</code> high productivity build system on top of <code>ninja</code>
xutils-dev	V:1, I:11	1466	<code>imake(1)</code> , <code>xmkmf(1)</code> , etc.

Tabela 12.16: List of build tool packages

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. 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.

Corra `make -p -f/dev/null` para ver as regras internas automáticas.

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

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

[Autotools](#) is a suite of programming tools designed to assist in making source code packages portable to many [Unix-like](#) systems.

- [Autoconf](#) is a tool to produce a shell script "configure" from "configure.ac".
 - "configure" is used later to produce "Makefile" from "Makefile.in" template.
- [Automake](#) is a tool to produce "Makefile.in" from "Makefile.am".
- [Libtool](#) is a shell script to address the software portability problem when compiling shared libraries from source code.

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
```

If you are not sure where files are installed, you should consider using `checkinstall(8)` from the `checkinstall` package, which provides a clean path for the uninstall. It now supports to create a Debian package with `"-D"` option.

12.6.3 Meson

The software build system has been evolving:

- [Autotools](#) on the top of [Make](#) has been the de facto standard for the portable build infrastructure since 1990s. This is extremely slow.
- [CMake](#) initially released in 2000 improved speed significantly but was still build on the top of inherently slow [Make](#).
- [Ninja](#) initially released in 2012 is meant to replace Make for the further improved build speed but is also designed to have its input files generated by a higher-level build system.
- [Meson](#) initially released in 2013 is the new popular and fast higher-level build system which uses [Ninja](#) as its backend.

See documents found at ["The Meson Build system"](#) and ["The Ninja build system"](#).

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.
 - `"http://www.foo.dom/cgi-bin/program.pl?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3"`
 - `"http://www.foo.dom/cgi-bin/program.py?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3"`
 - `"http://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 directamente o URL codificado <http://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:609, I:992	705	AWK → PERL	converter código-fonte de AWK para PERL: a2p(1)
f2c	V:0, I:5	442	FORTTRAN → C	converter código-fonte de FORTRAN 77 para C/C++: f2c(1)
intel2gas	V:0, I:0	178	intel → 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)`, and `sbuid(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 2.87)" (2021-11-03 08:37:36 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.

I have updated "Debian Reference (version 2)" to address new topics (Systemd, Wayland, IMAP, PipeWire, Linux kernel 5.10) and removed outdated topics (SysV init, CVS, Subversion, SSH protocol 1, Linux kernels before 2.5). References to Jessie 8 (2015-2020) release situation or older are mostly removed.

This "Debian Reference (version 2.87)" (2021-11-03 08:37:36 UTC) covers mostly Bullseye (=stable) and Bookworm (=testing) Debian releases.

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"
- "[Debian Tutorial](#)" por Havoc Pennington. (11 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.

- "[Debian Reference](#) (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

Many manual pages and info pages on the Debian system as well as upstream web pages and [Wikipedia](#) documents were used as the primary references to write this document. To the extent Osamu Aoki considered within the [fair use](#), many parts of them, especially command definitions, were used as phrase pieces after careful editorial efforts to fit them into the style and the objective of this document.

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 2.87)" (2021-11-03 08:37:36 UTC) é maioritariamente trabalho meu com excepção do descrito acima. Este foi também atualizado pelos contribuidores.

O Documento "Debian Reference (versão 2)" foi traduzido por Américo Monteiro [a_monteiro_AT_netcab.pt](#).

O autor, Osamu Aoki, agradece a todos os que ajudaram a tornar possível este documento.

A.3 Formato do documento

The source of the English original document is currently written in [DocBook](#) XML files. This Docbook XML source are converted to HTML, epub, plain text, PostScript, and PDF. (Some formats may be skipped for distribution.)