



debian

Debian Reference

Osamu Aoki

Copyright © 2013-2021 Osamu Aoki

Questa guida di riferimento Debian, Debian Reference (versione 2.87) (2021-11-03 08:37:36 UTC) è pensata per fornire un'ampia panoramica del sistema Debian in qualità di guida post-installazione per l'utente. Copre molti aspetti dell'amministrazione di sistema attraverso esempi di uso di comandi di shell, pensati per chi non è sviluppatore.

COLLABORATORI

	<i>TITOLO :</i> Debian Reference		
<i>AZIONE</i>	<i>NOME</i>	<i>DATA</i>	<i>FIRMA</i>
A CURA DI	Osamu Aoki	3 novembre 2021	

CRONOLOGIA DELLE REVISIONI

<i>POSIZIONE</i>	<i>DATA</i>	<i>DESCRIZIONE</i>	<i>NOME</i>

Indice

1	Tutorial GNU/Linux	1
1.1	Nozioni di base sulla console	1
1.1.1	Il prompt di shell	1
1.1.2	The shell prompt under GUI	2
1.1.3	L'account root	2
1.1.4	Il prompt di shell di root	3
1.1.5	Strumenti di amministrazione del sistema con interfaccia utente grafica	3
1.1.6	Console virtuali	3
1.1.7	Come uscire dal prompt dei comandi	3
1.1.8	Come spegnere il sistema	4
1.1.9	Ripristinare una console funzionante	4
1.1.10	Suggerimenti per pacchetti aggiuntivi per i principianti	4
1.1.11	Un account utente extra	5
1.1.12	Configurazione di sudo	5
1.1.13	Tempo di giocare	6
1.2	File system stile Unix	6
1.2.1	Nozioni di base sui file Unix	6
1.2.2	Aspetti tecnici del file system	8
1.2.3	Permessi del file system	8
1.2.4	Controllo dei permessi per i file appena creati: umask	10
1.2.5	Permessi per gruppi di utenti (gruppi)	11
1.2.6	Orari	12
1.2.7	Collegamenti	13
1.2.8	Pipe con nome (FIFO)	14
1.2.9	Socket	14
1.2.10	File di device	14
1.2.11	File di device speciali	15
1.2.12	procfs e sysfs	15
1.2.13	tmpfs	16
1.3	Midnight Commander (MC)	16

1.3.1	Personalizzazione di MC	17
1.3.2	Avvio di MC	17
1.3.3	Gestore dei file in MC	17
1.3.4	Trucchetti per la riga di comando di MC	17
1.3.5	L'editor interno di MC	18
1.3.6	Il visualizzatore interno di MC	18
1.3.7	Funzionalità di avvio automatico di MC	19
1.3.8	File system FTP virtuale di MC	19
1.4	Ambiente di lavoro di base in stile Unix	19
1.4.1	La shell di login	19
1.4.2	Personalizzare bash	20
1.4.3	Associazioni di tasti speciali	20
1.4.4	Mouse operations	21
1.4.5	Il paginatore	22
1.4.6	L'editor di testo	22
1.4.7	Impostare un editor di testi predefinito	22
1.4.8	Using vim	22
1.4.9	Registrare le attività della shell	23
1.4.10	Comandi Unix di base	24
1.5	Il semplice comando di shell	24
1.5.1	Esecuzione dei comandi e variabili d'ambiente	26
1.5.2	La variabile "\$LANG"	26
1.5.3	La variabile "\$PATH"	27
1.5.4	La variabile "\$HOME"	27
1.5.5	Opzioni della riga di comando	28
1.5.6	Glob della shell	28
1.5.7	Valore restituito dal comando	29
1.5.8	Sequenze tipiche di comandi e ridirezione della shell	29
1.5.9	Alias di comandi	31
1.6	Elaborazione di testo stile Unix	32
1.6.1	Strumenti Unix per il testo	32
1.6.2	Espressioni regolari	33
1.6.3	Espressioni di sostituzione	34
1.6.4	Sostituzione globale con espressioni regolari	34
1.6.5	Estrarre dati da file con tabelle di testo	35
1.6.6	Frammenti di script per comandi con pipe	36

2	Gestione dei pacchetti in Debian	38
2.1	Prerequisiti per la gestione dei pacchetti Debian	38
2.1.1	Configurazione dei pacchetti	38
2.1.2	Precauzioni base	39
2.1.3	Una vita di aggiornamenti senza fine	40
2.1.4	Nozioni di base sugli archivi Debian	41
2.1.5	Debian è al 100% software libero	44
2.1.6	Dipendenze dei pacchetti	45
2.1.7	Il flusso di eventi nella gestione dei pacchetti	46
2.1.8	Prima risposta a problemi di gestione dei pacchetti	47
2.2	Operazioni base per la gestione dei pacchetti	47
2.2.1	Confronti <code>apt - apt-get / apt-cache - aptitude</code>	47
2.2.2	Operazioni base per la gestione dei pacchetti dalla riga di comando	50
2.2.3	Uso interattivo di <code>aptitude</code>	50
2.2.4	Associazioni dei tasti per <code>aptitude</code>	51
2.2.5	Viste dei pacchetti in <code>aptitude</code>	51
2.2.6	Opzioni per i metodi di ricerca in <code>aptitude</code>	52
2.2.7	La struttura delle espressioni regolari di <code>aptitude</code>	53
2.2.8	Risoluzione delle dipendenze di <code>aptitude</code>	55
2.2.9	Registri delle attività sui pacchetti	55
2.3	Esempi di operazioni con <code>aptitude</code>	55
2.3.1	Elencare pacchetti in base alla corrispondenza del nome con espressioni regolari	55
2.3.2	Sfogliare le corrispondenze ad una espressione regolare	55
2.3.3	Eliminare completamente i pacchetti rimossi	55
2.3.4	Mettere ordine nello stato di installazione automatico/non automatico	56
2.3.5	Aggiornamento di tutto il sistema	56
2.4	Operazioni avanzate per la gestione dei pacchetti	58
2.4.1	Operazioni avanzate per la gestione dei pacchetti dalla riga di comando	58
2.4.2	Verifica dei pacchetti installati	58
2.4.3	Salvaguardarsi da problemi coi pacchetti	60
2.4.4	Cercare tra i metadati dei pacchetti	60
2.5	Aspetti tecnici della gestione dei pacchetti in Debian	60
2.5.1	Metadati degli archivi	60
2.5.2	File "Release" nella directory principale ed autenticità	60
2.5.3	File "Release" a livello di archivio	62
2.5.4	Recuperare i metadati per un pacchetto	62
2.5.5	Lo stato dei pacchetti per APT	63
2.5.6	Lo stato dei pacchetti per <code>aptitude</code>	63
2.5.7	Copie locali dei pacchetti scaricati	63

2.5.8	Nomi dei file di pacchetto di Debian	63
2.5.9	Il comando dpkg	64
2.5.10	Il comando update-alternatives	64
2.5.11	Il comando dpkg-statoverride	65
2.5.12	Il comando dpkg-divert	66
2.6	Ripristino da un sistema con problemi	66
2.6.1	Incompatibilità con vecchie configurazioni utente	66
2.6.2	Pacchetti diversi con file sovrapposti	66
2.6.3	Risolvere problemi negli script di pacchetto	67
2.6.4	Ripristinare con il comando dpkg	67
2.6.5	Ripristinare i dati sui pacchetti selezionati	68
2.7	Suggerimenti per la gestione dei pacchetti	68
2.7.1	Come scegliere i pacchetti Debian	68
2.7.2	Pacchetti da fonti mescolate di archivi	69
2.7.3	Modificare la versione candidata	70
2.7.4	Updates e Backports	72
2.7.5	Bloccare i pacchetti da installare perché "Raccomandati"	72
2.7.6	Usare e aggiornare <code>testing</code> con alcuni pacchetti da <code>unstable</code>	72
2.7.7	Usare e aggiornare <code>unstable</code> con alcuni pacchetti da <code>experimental</code>	74
2.7.8	Scaricare ed aggiornare automaticamente i pacchetti	74
2.7.9	Limitare l'uso di banda di APT per gli scaricamenti	75
2.7.10	Retrocessione di emergenza alla versione precedente	75
2.7.11	Chi ha caricato il pacchetto?	75
2.7.12	Il pacchetto equivs	76
2.7.13	Fare il port di un pacchetto nel sistema stabile	76
2.7.14	Server proxy per APT	77
2.7.15	Ulteriori letture sulla gestione dei pacchetti	77
3	Inizializzazione del sistema	78
3.1	Panoramica del processo di avvio	78
3.1.1	Stage 1: the UEFI	78
3.1.2	Stadio 2: il bootloader	79
3.1.3	Stadio 3: il mini-sistema Debian	80
3.1.4	Stadio 4: il normale sistema Debian	81
3.2	Init systemd	81
3.2.1	Il nome host	82
3.2.2	Il filesystem	83
3.2.3	Inizializzazione delle interfacce di rete	83
3.3	I messaggi del kernel	83

3.4	I messaggi di sistema	83
3.5	System management	84
3.6	Other system monitors	84
3.7	Personalizzare systemd	84
3.7.1	Socket activation	84
3.8	Il sistema udev	86
3.8.1	L'inizializzazione dei moduli del kernel	87
4	Authentication and access controls	88
4.1	Autenticazione Unix normale	88
4.2	Gestire le informazioni su account e password	90
4.3	Password buone	90
4.4	Creare password cifrate	91
4.5	PAM e NSS	91
4.5.1	File di configurazione letti da PAM e NSS	92
4.5.2	La moderna gestione centralizzata del sistema	92
4.5.3	"Perché GNU su non supporta il gruppo wheel"	93
4.5.4	Regole più stringenti per le password	93
4.6	Sicurezza dell'autenticazione	94
4.6.1	Password sicure in Internet	94
4.6.2	Secure Shell, shell sicura	94
4.6.3	Misure aggiuntive di sicurezza per Internet	95
4.6.4	Rendere sicura la password di root	95
4.7	Altri controlli sugli accessi	96
4.7.1	sudo	96
4.7.2	PolicyKit	96
4.7.3	Limitare l'accesso ad alcuni servizi server	96
4.7.4	Linux security features	97
5	Impostazione della rete	98
5.1	L'infrastruttura base di rete	98
5.1.1	Risoluzione dei nomi di host	98
5.1.2	Il nome dell'interfaccia di rete	100
5.1.3	L'intervallo degli indirizzi di rete per la LAN	101
5.1.4	Il supporto per i dispositivi di rete	101
5.2	La configurazione moderna della rete per il desktop	101
5.2.1	Strumenti grafici di configurazione della rete	102
5.3	La configurazione moderna della rete senza GUI	102
5.4	Configurazione della rete a basso livello	103

5.4.1	Comandi iproute2	103
5.4.2	Operazioni sicure a basso livello sulla rete	103
5.5	Ottimizzazione della rete	104
5.5.1	Trovare l'MTU ottimale	104
5.5.2	Ottimizzazione di TCP per la WAN	105
5.6	Infrastruttura netfilter	105
6	Applicazioni per la rete	107
6.1	Browser web	107
6.1.1	Configurazione del browser	108
6.2	Il sistema di posta	108
6.2.1	Nozioni di base sulla posta elettronica	109
6.2.2	Modern mail service limitation	109
6.2.3	Historic mail service expectation	110
6.2.4	Agente di trasporto della posta (MTA)	110
6.2.4.1	La configurazione di exim4	110
6.2.4.2	La configurazione di Postfix con SASL	112
6.2.4.3	La configurazione dell'indirizzo di posta	113
6.2.4.4	Operazioni di base degli MTA	114
6.3	Il server e le utilità per l'accesso remoto (SSH)	115
6.3.1	Nozioni di base su SSH	115
6.3.2	User name on the remote host	116
6.3.3	Connettersi senza password remote	116
6.3.4	Gestire client SSH estranei	116
6.3.5	Impostare ssh-agent	117
6.3.6	Sending a mail from a remote host	117
6.3.7	Forwarding della porta per tunnel SMTP/POP3	117
6.3.8	Spegnere il sistema remoto su SSH	117
6.3.9	Risolvere il problemi con SSH	118
6.4	Server ed utilità per la stampa	118
6.5	Altri server di rete	118
6.6	Altri client di rete	119
6.7	Diagnosi dei demoni di 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	125
7.5.1	Tipi di carattere base	125
7.5.2	Font rasterization	125
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	Appunti	129
8	I18N e L10N	131
8.1	La localizzazione	131
8.1.1	Logica alla base dell'uso della localizzazione UTF-8	131
8.1.2	La (ri)configurazione della localizzazione	132
8.1.3	Codifica per i nomi di file	133
8.1.4	Messaggi localizzati e documentazione tradotta	133
8.1.5	Effetti della localizzazione	133
8.2	L'input da tastiera	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	Il supporto per metodo di input con IBus	134
8.2.4	Un esempio per il giapponese	134
8.3	L'output sul display	135
8.4	Caratteri dell'Asia dell'est con larghezza ambigua	136
9	Suggerimenti per il sistema	137
9.1	The console tips	137
9.1.1	Registrare le attività della shell in modo pulito	137
9.1.2	Il programma screen	138
9.1.3	Navigating around directories	139
9.1.4	Readline wrapper	139
9.2	Personalizzare vim	139
9.2.1	Customizing vim with internal features	140
9.2.2	Customizing vim with external packages	140
9.3	Registrazione e presentazione di dati	141
9.3.1	Demoni di registro	141
9.3.2	Analizzatori di registro	141
9.3.3	Visualizzazione personalizzata di dati di testo	141

9.3.4	Visualizzazione personalizzata di date e orari	142
9.3.5	Output colorato per la shell	142
9.3.6	Comandi colorati	143
9.3.7	Registrare le attività dell'editor per ripetizioni complesse	143
9.3.8	Registrare l'immagine grafica di un'applicazione X	144
9.3.9	Registrare i cambiamenti a file di configurazione	144
9.4	Monitorare, controllare ed avviare l'attività dei programmi	144
9.4.1	Cronometrare un processo	144
9.4.2	Priorità di schedulazione	146
9.4.3	Il comando ps	146
9.4.4	Il comando top	146
9.4.5	Elencare i file aperti da un processo	146
9.4.6	Tenere traccia delle attività di un programma	146
9.4.7	Identificazione di processi in base a file o socket	147
9.4.8	Ripetere un comando ad intervalli costanti	147
9.4.9	Ripetere un comando su diversi file	147
9.4.10	Avviare un programma dalla GUI	148
9.4.11	Personalizzare i programmi da avviare	149
9.4.12	Uccidere un processo	151
9.4.13	Pianificare compiti da eseguire una volta sola	151
9.4.14	Pianificare compiti in modo regolare	151
9.4.15	Tasto Alt-SysRq	152
9.5	Suggerimenti per l'amministrazione del sistema	152
9.5.1	Chi è nel sistema?	152
9.5.2	Avvertire tutti gli utenti	153
9.5.3	Identificazione dell'hardware	153
9.5.4	Configurazione dell'hardware	153
9.5.5	Orario di sistema e hardware	154
9.5.6	Configurazione del terminale	154
9.5.7	L'infrastruttura audio	155
9.5.8	Disabilitare il salvaschermo	155
9.5.9	Disabilitare i bip sonori	155
9.5.10	Uso della memoria	156
9.5.11	Verifica della sicurezza e dell'integrità del sistema	156
9.6	Suggerimenti per l'archiviazione dei dati	157
9.6.1	Uso dello spazio su disco	157
9.6.2	Configurazione del partizionamento dei dischi	158
9.6.3	Accedere alle partizioni usando UUID	158
9.6.4	LVM2	159

9.6.5	Configurazione del file system	159
9.6.6	Creare file system e verificarne l'integrità	160
9.6.7	Ottimizzare il file system con opzioni di mount	160
9.6.8	Ottimizzare il file system tramite il superblocco	161
9.6.9	Ottimizzare il disco rigido	161
9.6.10	Ottimizzare le unità a stato solido	161
9.6.11	Usare SMART per prevedere danni ai dischi fissi	161
9.6.12	Specificare una directory per l'archiviazione di dati temporanei usando \$TMPDIR	162
9.6.13	Espandere lo spazio di archiviazione utilizzabile con LVM	162
9.6.14	Espandere lo spazio di archiviazione utilizzabile montando un'altra partizione	162
9.6.15	Espandere lo spazio di archiviazione utilizzabile montando un collegamento ad un'altra directory	162
9.6.16	Espandere lo spazio di archiviazione utilizzabile montando un'altra directory da sovrapporre	163
9.6.17	Espandere lo spazio di archiviazione utilizzabile usando collegamenti simbolici	163
9.7	Immagine del disco	163
9.7.1	Creare un file con un'immagine di disco	163
9.7.2	Scrivere direttamente sul disco	164
9.7.3	Montare un file con un'immagine di disco	164
9.7.4	Pulire un file con un'immagine di disco	165
9.7.5	Creare un file con immagine di disco vuoto	166
9.7.6	Creare un file con un'immagine ISO9660	166
9.7.7	Scrivere direttamente sul CD/DVD-R/RW	167
9.7.8	Montare un file con un'immagine ISO9660	167
9.8	I dati binari	167
9.8.1	Visualizzare e modificare dati binari	168
9.8.2	Manipolare file senza montare i dischi	168
9.8.3	Dati ridondanti	168
9.8.4	recupero di file dati ed analisi forensi	169
9.8.5	Suddividere un file grande in file più piccoli	169
9.8.6	Pulire il contenuto di file	169
9.8.7	File fittizi	170
9.8.8	Cancellare un intero disco fisso	170
9.8.9	Cancellare area inutilizzate di un disco fisso	170
9.8.10	De-cancellare file cancellati ma ancora aperti	171
9.8.11	Cercare tutti i collegamenti fisici	171
9.8.12	Consumo invisibile dello spazio su disco	172
9.9	Suggerimenti per la cifratura dei dati	172
9.9.1	Cifratura di dischi removibili con dm-crypt/LUKS	172
9.9.2	Cifratura della partizione di swap con dm-crypt	173
9.9.3	Montare dischi removibili con dm-crypt/LUKS	173

9.10	Il kernel	173
9.10.1	Parametri del kernel	174
9.10.2	Header del kernel	174
9.10.3	Compilare il kernel ed i moduli relativi	174
9.10.4	Compilare i sorgenti del kernel: il metodo raccomandato dal Team del Kernel di Debian	175
9.10.5	Driver per hardware e firmware	176
9.11	Sistema virtualizzato	176
9.11.1	Virtualization and emulation tools	177
9.11.2	Fasi del processo di virtualizzazione	178
9.11.3	Montare il file immagine di disco virtuale	178
9.11.4	Sistema chroot	179
9.11.5	Sistemi desktop multipli	179
10	Gestione dei dati	181
10.1	Condividere, copiare ed archiviare	181
10.1.1	Strumenti di archiviazione e compressione	182
10.1.2	Strumenti di copia e sincronizzazione	182
10.1.3	Esempi di invocazione per archivi	182
10.1.4	Esempi di invocazione per la copia	184
10.1.5	Esempi di invocazione per la selezione di file	185
10.1.6	Supporti di archiviazione	186
10.1.7	Supporti di archiviazione removibili	187
10.1.8	Scelta del file system per la condivisione di dati	188
10.1.9	Condividere dati attraverso una rete	189
10.2	Backup e ripristino	190
10.2.1	Suite con utilità di backup	191
10.2.2	Personal backup	192
10.3	Infrastruttura di sicurezza dei dati	192
10.3.1	Gestione delle chiavi per GnuPG	192
10.3.2	Usare GnuPG su file	194
10.3.3	Usare GnuPG con Mutt	194
10.3.4	Usare GnuPG con Vim	194
10.3.5	Somme di controllo MD5	196
10.4	Strumenti per la fusione di codice sorgente	196
10.4.1	Estrarre differenze da file sorgenti	196
10.4.2	Fondere aggiornamenti per file sorgenti	196
10.4.3	Interactive merge	196
10.5	Git	198
10.5.1	Configurazione del client Git	198
10.5.2	Basic Git commands	198
10.5.3	Git tips	199
10.5.4	Documenti di consultazione per Git	201
10.5.5	Other version control systems	201

11 Conversione di dati	203
11.1 Strumenti di conversione di dati testuali	203
11.1.1 Convertire un file di testo con iconv	203
11.1.2 Controllare se un file è in UTF-8 con iconv	205
11.1.3 Convertire nomi di file con iconv	205
11.1.4 Conversione del carattere di fine riga	205
11.1.5 Conversione di tabulazioni	206
11.1.6 Editor con auto-conversione	206
11.1.7 Estrazione del testo puro	207
11.1.8 Evidenziare e formattare dati in puro testo	208
11.2 Dati XML	208
11.2.1 Suggerimenti base per XML	208
11.2.2 Elaborazione XML	209
11.2.3 Estrazione di dati XML	211
11.2.4 The XML data lint	211
11.3 Impaginazione	211
11.3.1 Impaginazione roff	212
11.3.2 TeX/LaTeX	212
11.3.3 Fare una bella stampa di una pagina di manuale	213
11.3.4 Creare una pagina di manuale	213
11.4 Dati stampabili	213
11.4.1 Ghostscript	213
11.4.2 Unire due file PS o PDF	214
11.4.3 Utilità per dati stampabili	214
11.4.4 Stampare con CUPS	214
11.5 Conversione dei dati di posta	214
11.5.1 Nozioni di base sui dati di posta	216
11.6 Strumenti per dati grafici	216
11.7 Conversioni di dati vari	218
12 Programmazione	219
12.1 Script shell	219
12.1.1 Compatibilità con la shell POSIX	220
12.1.2 Parametri di shell	220
12.1.3 Costrutti condizionali della shell	222
12.1.4 Cicli di shell	222
12.1.5 La sequenza di elaborazione della riga di comando di shell	223
12.1.6 Programmi di utilità per script di shell	224
12.2 Scripting in interpreted languages	225

12.2.1	Debugging interpreted language codes	225
12.2.2	GUI program with the shell script	226
12.2.3	Pazzie con corti script Perl	226
12.3	Coding in compiled languages	227
12.3.1	C	227
12.3.2	Semplice programma in C (gcc)	228
12.3.3	Flex - un Lex migliorato	228
12.3.4	Bison - Yacc migliorato	229
12.4	Strumenti di analisi statica del codice	229
12.5	Debug	229
12.5.1	Esecuzione base di gdb	229
12.5.2	Fare il debug di pacchetti Debian	231
12.5.3	Ottenere un backtrace	232
12.5.4	Comandi gdb avanzati	232
12.5.5	Controllare le dipendenze dalle librerie	233
12.5.6	Dynamic call tracing tools	233
12.5.7	Fare il debug di errori X	233
12.5.8	Strumenti per rilevazione di memory leak	233
12.5.9	Disassemblatore di binari	233
12.6	Build tools	234
12.6.1	Make	234
12.6.2	Autotools	235
12.6.2.1	Compilare ed installare un programma	235
12.6.2.2	Disinstallare un programma	235
12.6.3	Meson	235
12.7	Web	236
12.8	Traduzione di codice sorgente	236
12.9	Creare pacchetti Debian	236
A	Appendice	238
A.1	Il labirinto Debian	238
A.2	Storia del copyright	238
A.3	Formato del documento	239

Elenco delle tabelle

1.1	Elenco di pacchetti con interessanti programmi in modalità testuale	4
1.2	Elenco di pacchetti con documentazione interessante	5
1.3	Elenco degli usi delle directory principali	7
1.4	Elenco dei valori possibili per il primo carattere nell'output di "ls -l"	9
1.5	Permessi in notazione numerica per i comandi chmod(1)	10
1.6	Esempi di valori di umask	11
1.7	Elenco dei principali gruppi forniti dal sistema per accesso ai file	11
1.8	Elenco dei principali gruppi forniti dal sistema per l'esecuzione di particolari comandi	12
1.9	Elenco dei tipi di data	12
1.10	Elenco di file dei device speciali	16
1.11	Le associazioni dei tasti di MC	18
1.12	La reazione al tasto Invio in MC	19
1.13	Elenco di programmi shell	19
1.14	Elenco di associazioni di tasti per bash	21
1.15	List of mouse operations and related key actions on Debian	21
1.16	List of basic Vim key strokes	23
1.17	Elenco di comandi Unix di base	25
1.18	Le 3 parti del valore di localizzazione	26
1.19	Elenco di localizzazioni raccomandate	27
1.20	Elenco di valori di "\$HOME"	28
1.21	Modelli di glob della shell	28
1.22	Codici di uscita dei comandi	29
1.23	Idiomi per comandi di shell	30
1.24	Descrittori di file predefiniti	31
1.25	Metacaratteri per BRE e ERE	33
1.26	L'espressione di sostituzione	34
1.27	Elenco di frammenti di script per comandi con pipe	37
2.1	Elenco degli strumenti Debian di gestione dei pacchetti	39
2.2	Elenco dei siti con l'archivio Debian	42

2.3	Elenco delle aree dell'archivio Debian	42
2.4	Relazione tra suite e nome in codice	43
2.5	Elenco dei siti web importanti per la risoluzione di problemi con un pacchetto specifico	47
2.6	Operazioni base di gestione dei pacchetti dalla riga di comando usando <code>aptitude(8)</code> e <code>apt-get(8)/apt-cache(8)</code>	49
2.7	Opzioni degne di nota per il comando <code>aptitude(8)</code>	50
2.8	Elenco delle associazioni di tasti per <code>aptitude</code>	51
2.9	Elenco delle viste di <code>aptitude</code>	52
2.10	Organizzazione delle viste standard dei pacchetti	52
2.11	Elenco delle regole per espressioni regolari di <code>aptitude</code>	54
2.12	File di registro per le attività sui pacchetti	55
2.13	Elenco delle operazioni avanzate per la gestione dei pacchetti	59
2.14	Il contenuto dei metadati dell'archivio Debian	61
2.15	La struttura dei nomi dei pacchetti Debian	63
2.16	I caratteri utilizzabili all'interno di ciascuna porzione del nome dei pacchetti Debian	64
2.17	I file degni di nota creati da <code>dpkg</code>	65
2.18	Elenco di valori di priorità di pin che esemplificano la tecnica d'uso dei pin di APT.	71
2.19	Elenco degli strumenti proxy specifici per l'archivio Debian	77
3.1	Elenco di bootloader	79
3.2	The meaning of the menu entry of the above part of <code>/boot/grub/grub.cfg</code>	80
3.3	Elenco di utilità di avvio per il sistema Debian	81
3.4	Elenco dei livelli di errore del kernel	83
3.5	List of typical <code>journalctl</code> command snippets	84
3.6	List of typical <code>systemctl</code> command snippets	85
3.7	List of other monitoring command snippets under <code>systemd</code>	86
4.1	I 3 importanti file di configurazione per <code>pam_unix(8)</code>	88
4.2	Il contenuto della seconda voce di <code>"/etc/passwd"</code>	89
4.3	Elenco di comandi per gestire informazioni su account	90
4.4	Elenco di strumenti per generare password	91
4.5	Elenco dei pacchetti degni di nota per i sistemi PAM e NSS	91
4.6	Elenco di file di configurazione letti da PAM e NSS	92
4.7	Elenco di servizi e porte sicuri e non sicuri	94
4.8	Elenco di strumenti per fornire misure aggiuntive di sicurezza	95
5.1	Elenco degli strumenti di configurazione della rete	99
5.2	Elenco di intervalli di indirizzi di rete	101
5.3	Tabella di traduzione dai comandi obsoleti <code>net-tools</code> ai nuovi comandi <code>iproute2</code>	103
5.4	Elenco di comandi di rete a basso livello	103
5.5	Elenco degli strumenti di ottimizzazione della rete	104

5.6	Linee guida di base per il valore di MTU ottimale	105
5.7	Elenco di strumenti per firewall	106
6.1	Elenco di browser web	107
6.2	Elenco di pacchetti con plugin per browser	108
6.3	Elenco di programmi di posta (MUA)	109
6.4	List of basic mail transport agent related packages	111
6.5	Elenco delle pagine di manuale di Postfix importanti	112
6.6	Elenco dei file di configurazione correlati all'indirizzo di posta	113
6.7	Elenco di operazioni base degli MTA	114
6.8	Elenco dei server e delle utilità per l'accesso remoto	115
6.9	Elenco dei file di configurazione per SSH	116
6.10	Elenco di esempi di avvio di client SSH	116
6.11	Elenco di client SSH per altre piattaforme	117
6.12	Elenco di server e utilità di stampa	118
6.13	Elenco di altri server applicativi di rete	119
6.14	Elenco di altri client applicativi di rete	120
6.15	Elenco di RFC popolari	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	126
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	Elenco di metodi di connessione al server 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	Elenco di associazioni di tasti per screen	139
9.3	Information on the initialization of vim	141
9.4	Elenco di analizzatori del registro di sistema	142
9.5	Display examples of time and date for the "ls -l" command with the time style value	142
9.6	Elenco di strumenti di manipolazione di immagini grafiche	144
9.7	List of packages which can record configuration history	144
9.8	Elenco di strumenti per monitorare e controllare l'attività dei programmi.	145
9.9	Elenco di valori di nice per la priorità di schedulazione	145

9.10	Elenco degli stili per il comando <code>ps</code>	146
9.11	Elenco dei segnali usati comunemente con il comando <code>kill</code>	150
9.12	List of notable SAK command keys	152
9.13	Elenco di strumenti per l'identificazione dell'hardware	153
9.14	Elenco di strumenti di configurazione dell'hardware	153
9.15	Elenco di pacchetti relativi all'audio	155
9.16	Elenco di comandi per disabilitare il salvaschermo	156
9.17	Elenco di dimensioni della memoria riportate	156
9.18	Elenco di strumenti per verificare la sicurezza e l'integrità del sistema	157
9.19	Elenco di pacchetti di gestione delle partizioni dei dischi	158
9.20	Elenco di pacchetti di gestione dei file system	160
9.21	Elenco di pacchetti che visualizzano e modificano dati binari	168
9.22	Elenco di pacchetti per manipolare file senza montare i dischi	168
9.23	Elenco di strumenti per aggiungere dati ridondanti a file	168
9.24	Elenco di pacchetti per recupero di file dati ed analisi forensi.	169
9.25	Elenco di utilità per la cifratura dei dati	172
9.26	Elenco di pacchetti chiave da installare per la ricompilazione del kernel in un sistema Debian	174
9.27	Elenco di strumenti di virtualizzazione	177
10.1	Elenco di strumenti di archiviazione e compressione	183
10.2	Elenco di strumenti di copia e sincronizzazione	184
10.3	Elenco di possibili scelte per il file system di dispositivi di archiviazione removibili con scenari di uso tipici	188
10.4	Elenco dei servizi di rete da scegliere in base allo scenario di uso tipico	190
10.5	Elenco di suite con utilità di backup	192
10.6	Elenco di strumenti per l'infrastruttura di sicurezza dei dati	193
10.7	Elenco di comandi per GNU Privacy Guard per la gestione delle chiavi	193
10.8	Elenco dei significati dei codici di fiducia	194
10.9	Elenco di comandi GNU Privacy Guard per file	195
10.10	Elenco di strumenti per la fusione di codice sorgente	197
10.11	Elenco di pacchetti e comandi relativi a Git	198
10.12	Main Git commands	199
10.13	Git tips	200
10.14	List of other version control system tools	202
11.1	Elenco di strumenti di conversione di dati testuali	203
11.2	Elenco dei valori delle codifiche e loro uso	204
11.3	Elenco di stili per EOL per differenti piattaforme	206
11.4	Elenco di comandi di conversione di TAB dai pacchetti <code>bsdmainutils</code> e <code>coreutils</code>	206
11.5	Elenco di strumenti per estrarre dati in testo puro	207

11.6	Elenco di strumenti per evidenziare dati in testo puro	208
11.7	Elenco di entità predefinite per XML	209
11.8	Elenco di strumenti XML	210
11.9	Elenco di strumenti DSSSL	210
11.10	Elenco di strumenti di estrazione di dati XML	211
11.11	Elenco di strumenti per belle stampe XML	211
11.12	Elenco di strumenti per impaginazione	212
11.13	Elenco di pacchetti che aiutano a creare una pagina man	213
11.14	Elenco di interpreti PostScript Ghostscript	214
11.15	Elenco di utilità per dati stampabili	215
11.16	Elenco di pacchetto che aiutano a convertire dati di posta	215
11.17	Elenco di strumenti per dati grafici	217
11.18	Elenco di strumenti di conversione di dati vari	218
12.1	Elenco di bashismi tipici	220
12.2	Elenco di parametri di shell	221
12.3	Elenco di espansioni di parametri di shell	221
12.4	Elenco di sostituzioni chiave di parametri di shell	221
12.5	Elenco di operatori per paragonare file in espressioni condizionali	222
12.6	Elenco di operatori per paragonare stringhe in espressioni condizionali	223
12.7	Elenco di pacchetti contenenti piccoli programmi di utilità per script di shell	224
12.8	List of interpreter related packages	225
12.9	List of dialog programs	226
12.10	List of compiler related packages	227
12.11	Elenco di generatori di parser LALR compatibili con Yacc	229
12.12	Elenco di strumenti per l'analisi statica del codice	230
12.13	List of debug packages	230
12.14	Elenco di comandi gdb avanzati	232
12.15	Elenco di strumenti per rilevazione di memory leak	233
12.16	List of build tool packages	234
12.17	Elenco di variabili automatiche di make	234
12.18	Elenco di espansioni delle variabili di make	234
12.19	Elenco di strumenti per la traduzione di codice sorgente	236

Sommario

Questo testo è libero; lo si può redistribuire e/o modificare nei termini della GNU General Public License in qualsiasi versione aderente alle Linee guida Debian per il Software Libero (DFSG).

Prefazione

Questa guida di riferimento [Debian Reference \(versione 2.87\)](#) (2021-11-03 08:37:36 UTC) è pensata per fornire un'ampia panoramica dell'amministrazione di un sistema Debian in qualità di guida post-installazione per l'utente.

Si rivolge ad un lettore che abbia voglia di imparare gli script di shell, ma che non è pronto a leggere tutti i sorgenti C per scoprire come funzionano il sistema [GNU/Linux](#).

Per istruzioni sull'installazione, vedere:

- [Guida all'installazione di Debian GNU/Linux per l'attuale sistema stabile](#)
- [Guida all'installazione di Debian GNU/Linux per l'attuale sistema testing](#)

Note legali

Non viene fornita alcuna garanzia. Tutti i marchi registrati appartengono ai rispettivi proprietari.

Il sistema Debian è, di per sé, un'entità in continuo movimento; ciò rende difficile alla sua documentazione essere aggiornata e corretta. Sebbene come base per questo testo sia stata usata la versione "unstable" al momento della sua stesura, alcuni contenuti potrebbero essere obsoleti al momento della sua lettura.

Si consideri questo documento come un riferimento secondario. Esso non sostituisce nessuna guida autorevole. L'autore e i collaboratori non si assumono responsabilità per le conseguenze di errori, omissioni o ambiguità presenti in questo documento.

Cosa è Debian

Il [Progetto Debian](#) è un'associazione di individui che hanno fatto causa comune per creare un sistema operativo libero. La sua distribuzione è caratterizzata dai seguenti aspetti.

- Impegno per la libertà del software: [Il Contratto sociale Debian e le Linee Guida Debian per il Software Libero \(DFSG\)](#).
- Sforzo volontario non pagato, distribuito, basato su Internet <https://www.debian.org>.
- Vasto numero di pacchetti software di alta qualità pre-compilati.
- Attenzione alla stabilità e alla sicurezza, con facile accesso agli aggiornamenti di sicurezza.
- Attenzione ad aggiornamenti senza problemi ai pacchetti più recenti dei software negli archivi `unstable` e `testing`.
- Vasto numero di architetture hardware supportate.

I vari pezzi di Software Libero in Debian provengono da [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 molti altri progetti software liberi indipendenti. Debian integra questa diversità del Software Libero in un unico sistema.

Informazioni su questo documento

Linee guida

Nella stesura di questo documento sono state seguite le seguenti linee guida.

- Fornire una panoramica ed evitare casi particolari. (**Vista d'insieme.**)
- Mantenere il testo **conciso e semplice**.
- Non reinventare la ruota. (Usare rimandi alle **risorse esistenti.**)
- Concentrarsi sugli strumenti senza interfaccia utente grafica e sulla console. (Usare **esempi shell.**)
- Essere obiettivi. (Usare [popcon](#), ecc.)

Suggerimento

Si è cercato di spiegare aspetti gerarchici e i più bassi livelli di funzionamento del sistema.

Prerequisiti



avvertimento

Il lettore deve fare lo sforzo di cercare risposte da solo andando oltre questa documentazione. Questo documento fornisce solo degli efficaci punti di partenza.

Il lettore deve cercare le soluzioni da solo dalle fonti primarie.

- [The Debian Administrator's Handbook](#)
- Il sito Debian all'indirizzo <https://www.debian.org> per le informazioni generiche.
- La documentazione nella directory `"/usr/share/doc/nome_pacchetto"`.
- La **pagina man** in stile Unix: `"dpkg -L nome_pacchetto |grep '/man/man.*/'"`.
- La **pagina info** in stile GNU: `"dpkg -L nome_pacchetto |grep '/info/'"`.
- Le segnalazioni di bug: http://bugs.debian.org/nome_pacchetto .
- Il Debian Wiki all'indirizzo <https://wiki.debian.org/> per gli aspetti mutevoli e gli argomenti specifici.
- Gli HOWTO del progetto di documentazione di Linux TLDP (The Linux Documentation Project), all'indirizzo <http://tldp.org/> .
- Le specifiche Single UNIX Specification dal sito web di The UNIX System di Open Group, all'indirizzo <http://www.unix.org/>.
- L'enciclopedia libera Wikipedia all'indirizzo <https://www.wikipedia.org/> .

Nota

Per una documentazione dettagliata può essere necessario installare il corrispondente pacchetto di documentazione il cui nome ha il suffisso `"-doc"` .

Convenzioni

Questo documento fornisce informazioni attraverso il seguente stile semplificato di rappresentazione con esempi di comandi per la shell `bash(1)`.

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

Questi prompt di shell distinguono gli account usati e corrispondono a ciò che si ottiene impostando le variabili d'ambiente in questo modo: `"PS1='\$'"` e `"PS2=' '"`. Questi valori sono stati scelti per aumentare la leggibilità del documento e non sono quelli tipici di una reale installazione del sistema.

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

Si veda il significato delle variabili d'ambiente `"$PS1"` e `"$PS2"` in `bash(1)`.

Le **azioni** che gli amministratori di sistema devono eseguire sono scritte all'infinito, ad esempio "Premere il tasto Invio dopo ogni stringa di comando digitata nella shell."

La colonna **descrizione** ed altre simili in tabelle possono contenere un **sintagma nominale** che segue le [convenzioni per le descrizioni brevi dei pacchetti](#) e che perde l'articolo iniziale quali "un" ed "il". Le colonne possono, in alternativa, contenere una verbo al presente seguendo la convenzione usata nella descrizione breve dei comandi nelle pagine man. Questa scelta potrà apparire strana per qualcuno, ma fa parte della scelta deliberata dell'autore di mantenere più semplice possibile lo stile di questa documentazione. In entrambi i casi l'iniziale non è maiuscola e manca il punto finale seguendo le dette convenzioni per le descrizioni brevi.

Nota

I nomi propri, inclusi i nomi dei comandi mantengono la corretta iniziale maiuscola/minuscola indipendentemente dalla loro posizione.

Un **breve comando** citato all'interno di un paragrafo di testo viene reso dal carattere a spaziatura fissa racchiuso tra virgolette doppie come in `"aptitude safe-upgrade"`.

Dati testuali da un file di configurazione citati all'interno di un paragrafo di testo vengono resi dal carattere a spaziatura fissa racchiuso tra virgolette doppie come in `"deb-src"`.

Un **comando** viene reso dall'uso del carattere a spaziatura fissa per il suo nome, seguito eventualmente dal numero di sezione della sua pagina man tra parentesi, come in `bash(1)`. Ci si senta incoraggiati ad ottenere ulteriori informazioni digitando quanto segue.

```
$ man 1 bash
```

Una **pagina man** viene resa dall'uso del carattere a spaziatura fissa per il suo nome seguito dal numero della sua sezione nelle pagine man racchiuso tra parentesi, come in `sources.list(5)`. Ci si senta incoraggiati ad ottenere ulteriori informazioni digitando quanto segue.

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

Una **pagina info** viene resa dal suo comando corrispondente in carattere a spaziatura fissa racchiuso tra virgolette doppie, come in `"info make"`. Ci si senta incoraggiati ad ottenere ulteriori informazioni digitando quanto segue.

```
$ info make
```

Un **nome di file** viene reso dal carattere a spaziatura fissa tra virgolette doppie, come in `"/etc/passwd"`. Per quanto riguarda i file di configurazione ci si senta incoraggiati ad ottenere ulteriori informazioni digitando quanto segue.

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

Un **nome di directory** viene reso dal carattere a spaziatura fissa tra virgolette doppie, come in «/etc/apt/». Ci si senta incoraggiati ad esplorare il suo contenuto digitando quanto segue.

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

Un **nome di pacchetto** viene reso dal carattere a spaziatura fissa, come in vim. Ci si senta incoraggiati ad ottenere ulteriori informazioni digitando quanto segue.

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

La posizione di una **documentazione** può essere indicata con il nome di file in carattere a spaziatura fissa tra virgolette doppie, come in "/usr/share/doc/base-passwd/users-and-groups.txt.gz" e "/usr/share/doc/base-passwd/users-and-groups.html" oppure dal suo **URL**, come in <https://www.debian.org>. Ci si senta incoraggiati a leggere la documentazione digitando quanto segue.

```
$ 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"
```

Una **variabile d'ambiente** viene resa dal suo nome preceduto dal segno "\$" in carattere a spaziatura fissa tra virgolette doppie, come in "\$TERM". Ci si senta incoraggiati ad ottenere il suo valore attuale digitando quanto segue.

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

Popcon

I dati da [popcon](#) sono usati come misura oggettiva della popolarità di ciascun pacchetto. Sono stati scaricati il 2021-08-14 17:04:37 UTC e contengono i dati totali inviati da 196036 rapporti su 183707 pacchetti binari e 26 architetture.

Nota

Notare che l'archivio amd64 unstable contiene attualmente solo 63210 pacchetti. I dati di popcon contengono rapporti inviati da molte installazioni su sistemi vecchi.

Il numero popcon, preceduto da "V:" che sta per "voti", è calcolato così: "1000 * (rapporti popcon inviati con il pacchetto eseguito di recente sul PC)/(numero totale di rapporti popcon inviati)".

Il numero popcon, preceduto da "I:" che sta per "installazioni", è calcolato così: "1000 * (rapporti popcon inviati con il pacchetto installato sul PC)/(numero totale di rapporti popcon inviati)".

Nota

I numeri popcon non dovrebbero essere considerati una misura assoluta dell'importanza dei pacchetti. Ci sono molti fattori che possono influenzare le statistiche. Per esempio, alcuni sistemi che partecipano a popcon possono avere montato directory quali "/bin" con l'opzione "noatime" per migliorare le prestazioni di sistema, influenzando di fatto negativamente i "voti" per quel sistema.

La dimensione dei pacchetti

Anche i dati sulla dimensione dei pacchetti sono usati come misura oggettiva per ciascun pacchetto. Sono basati sulla "Installed-Size" riportata dal comando "apt-cache show" o "aptitude show" (attualmente sull'architettura amd64 per la versione unstable). La dimensione è riportata in KiB ([Kibibyte](#) = unità di 1024 byte).

Nota

Una piccola dimensione per un pacchetto può indicare che esso è, nella versione unstable un pacchetto fittizio che installa, attraverso le dipendenze, altri pacchetti con contenuti sostanziosi. Il pacchetto fittizio permette una transizione facile o una suddivisione di pacchetto.

Nota

Una dimensione di pacchetto seguita da "(*)" indica che il pacchetto non è presente nella versione unstable ed è stata quindi usata la dimensione del pacchetto in experimental al suo posto.

Segnalazioni di bug in questo documento

Se si trovano errori in questo documento segnalare i bug nel pacchetto `debian-reference` usando `reportbug(1)`. Se possibile, includere i suggerimenti di correzione come "diff -u" alla versione in puro testo o ai sorgenti.

Alcuni promemoria per i nuovi utenti

Alcuni promemoria per i nuovi utenti:

- Fare il backup dei propri dati
 - Mantenere sicure le proprie password e chiavi di sicurezza
 - [KISS \(keep it simple stupid\)](#), cioè mantenere le cose semplici
 - Non iper-progettare il proprio sistema
 - Leggere i propri file di log
 - Il **PRIMO** errore è quello che conta
 - [RTFM \(read the fine manual\)](#), cioè leggere il manuale
 - Cercare in Internet prima di fare domande
 - Non lavorare come root quando non è necessario
 - Non manomettere il sistema di gestione dei pacchetti
 - Non digitare nulla di cui non si capisce il significato
 - Non cambiare i permessi dei file (prima di una completa revisione della sicurezza)
 - Non lasciare la shell di root fino a che non si sono **TESTATE** le proprie modifiche
 - Avere sempre un supporto di avvio alternativo (chiavetta USB, CD, ...)
-

Alcune citazioni per i nuovi utenti

Ecco alcune interessanti citazioni dalla mailing list di Debian che potrebbero illuminare i nuovi utenti.

- "Questo è Unix. Ti dà abbastanza corda da impiccarti." --- Miquel van Smoorenburg <miquels at cistron.nl>
- "Unix È amichevole... è solo molto selettivo riguardo i suoi amici." --- Tollef Fog Heen <tollef at add.no>

Wikipedia ha un articolo (in inglese) sulla "[Filosofia Unix](#)" che elenca citazioni interessanti.

Capitolo 1

Tutorial GNU/Linux

Penso che imparare un sistema informatico sia come imparare una lingua straniera. Anche se le guide e la documentazione sono utili, si deve fare pratica diretta. Per aiutare il lettore a iniziare dolcemente, ho elaborato alcuni punti base.

Il potente design di [Debian GNU/Linux](#) deriva dal sistema operativo [Unix](#), cioè un sistema operativo [multiutente](#) e [multitasking](#). Bisogna imparare a sfruttare la potenza di queste caratteristiche e le somiglianze tra Unix e GNU/Linux.

Non bisogna scappare dai testi pensati per Unix e affidarsi solamente a testi su GNU/Linux dato che in questo modo ci si priva di molte informazioni utili.

Nota

Se si è usato tramite strumenti a riga di comando un sistema [*nix](#) per un certo tempo, probabilmente si sa già tutto ciò che viene spiegato qui. Si usi questa sezione come un ripasso e per consultazioni.

1.1 Nozioni di base sulla console

1.1.1 Il prompt di 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 Sezione [1.1.6](#) below for more).

```
foo login:
```

Al prompt di login digitare il proprio nome utente, ad esempio `pinguino` e premere il tasto Invio, poi digitare la propria password e premere di nuovo Invio.

Nota

Nella tradizione Unix, il nome utente e la password di un sistema Debian distinguono le lettere maiuscole dalle minuscole. Il nome utente di solito viene scelto usando solo lettere minuscole. Il primo account utente viene normalmente creato durante l'installazione. Account utente aggiuntivi possono essere creati da root con `adduser(8)`.

Il sistema si presenta con il messaggio di benvenuto memorizzato in `"/etc/motd"` (Message Of The Day, messaggio del giorno) e fornisce un prompt dei comandi.

```
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:~$
```

Si è ora nella [shell](#). La shell interpreta i comandi dell'utente.

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.

In altri sistemi Desktop (come `fluxbox`), potrebbe non esserci un punto evidente per l'apertura del menu. Se si è in questa condizione, provare a fare clic, con il tasto destro, sullo sfondo della schermata del desktop e sperare che appaia un menu.

1.1.3 L'account root

L'account root viene anche indicato come [superutente](#) o utente privilegiato. Da questo account si possono eseguire i seguenti compiti amministrativi.

- Leggere, scrivere e cancellare qualsiasi file sul sistema indipendentemente dai suoi permessi.
- Impostare il proprietario e i permessi di qualunque file sul sistema.
- Impostare la password di qualsiasi utente non privilegiato nel sistema.
- Fare il login in qualsiasi account senza la password.

Questi poteri illimitati dell'account root rendono necessario essere prudenti e responsabili nel loro uso.



avvertimento

Non comunicare mai la password di root ad altri.

Nota

I permessi di un file (inclusi i device hardware come CD-ROM, ecc. che sono nient'altro che un altro file per il sistema Debian) lo possono rendere inutilizzabile o inaccessibile per gli utenti non root. Benché usare l'account root sia un metodo veloce per affrontare questo tipo di situazione, la sua soluzione dovrebbe essere l'impostazione degli appropriati permessi e gruppi proprietari per il file (vedere Sezione [1.2.3](#)).

1.1.4 Il prompt di shell di root

Ecco alcuni metodi di base per ottenere il prompt di shell per root usando la password di root.

- Inserire `root` al prompt di login a caratteri.
- Digitare `su -l` al prompt di shell di un utente qualsiasi.
 - L'ambiente dell'utente attuale non viene in questo caso preservato.
- Digitare `su` al prompt di shell di un utente qualsiasi.
 - L'ambiente dell'utente attuale viene in questo caso parzialmente preservato.

1.1.5 Strumenti di amministrazione del sistema con interfaccia utente grafica

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 Sezione 1.1.4 and Sezione 7.8.



avvertimento

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)`.

Non eseguire mai in X Window programmi con interfaccia utente grafica non fidati da remoto quando sono visualizzate informazioni critiche, dato che potrebbero spiare lo schermo X.

1.1.6 Console virtuali

Nel sistema Debian standard, ci sono sei console a caratteri [in stile VT100](#) disponibili tra cui ci si può spostare per avviare una shell di comando direttamente sull'host Linux. A meno che non si sia in un ambiente con interfaccia grafica, si può passare da una console virtuale all'altra usando simultaneamente il `tasto_Alt_sinistro` e uno dei tasti `F1` — `F6`. Ogni console a caratteri permette un login indipendente nell'account e offre un ambiente multiutente. L'ambiente multiutente è una bellissima caratteristica di Unix e ci si può abituare presto a dipendere da esso.

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 Come uscire dal prompt dei comandi

Si digita `Ctrl-D`, cioè il `tasto_Ctrl_sinistro` e il `tasto_d` vengono premuti contemporaneamente al prompt dei comandi per chiuderà l'attività della shell. Se si è nella console a caratteri, in questo modo si ritorna al prompt di login. Anche se questi caratteri di controllo vengono indicati con le lettere maiuscole, come "control D", non è necessario premere il tasto Maiusc. Per indicare `Ctrl-D` viene anche usata l'espressione abbreviata `^D`. In alternativa si può digitare "exit".

In quest'ultimo modo, se si è in un emulatore di terminale (`x-terminal-emulator(1)`), lo si può chiudere.

1.1.8 Come spegnere il sistema

Esattamente come ogni altro SO moderno in cui le operazioni su file comportano [una cache dei dati](#) in memoria per migliorare le prestazioni, il sistema Debian ha bisogno della appropriata procedura di spegnimento prima che l'alimentazione possa essere staccata in modo sicuro. Questo serve a preservare l'integrità dei file, forzando la scrittura su disco di tutti i cambiamenti avvenuti in memoria. Se è disponibile il controllo software dell'alimentazione, la procedura di spegnimento automaticamente toglie l'alimentazione al sistema. (In caso contrario può essere necessario tener premuto per alcuni secondi il tasto di accensione/spegnimento.)

Si può spegnere il sistema dalla riga di comando nella normale modalità multiutente.

```
# shutdown -h now
```

Si può spegnere il sistema dalla riga di comando nella modalità single-user.

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

Vedere Sezione [6.3.8](#).

1.1.9 Ripristinare una console funzionante

Quando, dopo aver fatto qualcosa di strano come `cat un-qualche-file-binario`, lo schermo impazzisce digitare `reset` al prompt dei comandi. Mentre lo si digita il comando potrebbe non essere visualizzato. Si può anche usare `clear` per pulire lo schermo.

1.1.10 Suggerimenti per pacchetti aggiuntivi per i principianti

Sebbene anche l'installazione minima del sistema Debian, senza alcun ambiente desktop, fornisca le funzionalità Unix di base è una buona idea installare con `apt-get(8)` alcuni pacchetti aggiuntivi per la riga di comando e i terminali a caratteri basati su curses, come `mc` e `vim` con cui i principianti possono fare i primi passi. Eseguire i comandi seguenti.

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

Se questi pacchetti sono già installati, nessun nuovo pacchetto sarà installato.

pacchetto	popcon	dimensione	descrizione
mc	V:55, I:234	1492	Gestore di file testuale a tutto schermo
sudo	V:597, I:809	4589	Programma per garantire privilegi di root limitati agli utenti
vim	V:102, I:404	3286	Editor di testi Unix Vi IMproved, un editor di testi per programmatori (versione standard)
vim-tiny	V:57, I:969	1574	Editor di testi Unix Vi IMproved, un editor di testi per programmatori (versione compatta)
emacs-nox	V:4, I:19	18364	Emacs dal progetto GNU, l'editor di testi estensibile basato su Lisp
w3m	V:25, I:252	2367	Browser Web testuali
gpm	V:11, I:16	540	Taglia e Incolla in stile Unix nella console testuale (demone)

Tabella 1.1: Elenco di pacchetti con interessanti programmi in modalità testuale

Potrebbe essere una buona idea leggere un po' di documentazione.

Si possono installare alcuni di questi pacchetti usando i comandi seguenti.

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

pacchetto	popcon	dimensione	descrizione
doc-debian	I:850	166	Documentazione del Progetto Debian, (FAQ Debian) ed altri documenti
debian-policy	I:29	4306	Manuale Debian Policy e documenti correlati
developers-reference	I:6	1916	Linee guida ed informazioni per gli sviluppatori Debian
debmake-doc	I:0	10989	Guide for Debian Maintainers
debian-history	I:1	4285	Storia del Progetto Debian
debian-faq	I:846	817	FAQ Debian

Tabella 1.2: Elenco di pacchetti con documentazione interessante

1.1.11 Un account utente extra

Se non si vuole usare il proprio account utente principale per le esercitazioni descritte in seguito, si può creare un account utente per esercitarsi, ad esempio `pesce` con il comando seguente.

```
# adduser fish
```

Rispondere a tutte le domande.

In questo modo si crea un nuovo account chiamato `pesce`. Dopo aver fatto pratica si può rimuovere questo account e la sua directory home digitando quanto segue.

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

1.1.12 Configurazione di sudo

Per la tipica postazione di lavoro di un unico utente, come il sistema desktop Debian su un PC portatile, è frequente l'uso di una semplice configurazione di `sudo(8)`, come qui di seguito descritto, per permettere all'utente non privilegiato, ad esempio `pinguino`, di ottenere privilegi di amministratore con la sola propria password, senza quella dell'utente `root`.

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

In alternativa è uso comune usare la configurazione seguente per permettere all'utente non privilegiato, ad esempio `pinguino`, di ottenere privilegi di amministratore senza alcuna password.

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

Questo trucco dovrebbe essere usato esclusivamente per le postazioni di lavoro con un solo utente, in cui l'utente è anche l'amministratore.



avvertimento

Non impostare account per utenti regolari su postazioni di lavoro multi-utente in questo modo perché ciò sarebbe un grosso problema per la sicurezza del sistema.



Attenzione

Nell'esempio precedente la password e l'account di `pinguino` richiedono altrettanta protezione della password e account dell'utente `root`.

I privilegi di amministrazione in questo contesto sono forniti a qualcuno che è autorizzato ad eseguire i compiti di amministrazione del sistema sulla macchina. Non dare mai tali privilegi ad un manager del dipartimento di Amministrazione della propria azienda o al proprio capo, a meno che non siano autorizzati e capaci.

Nota

Per fornire privilegi di accesso a specifici device e file, si dovrebbe considerare l'uso dei **gruppi** per dare un accesso limitato, invece di usare i privilegi di root attraverso sudo(8).

Con una configurazione più attenta e precisa, sudo(8) può garantire privilegi amministrativi limitati ad altri utenti in un sistema condiviso, senza rendere nota la password di root. Questo può aiutare la tracciabilità su host con più di un amministratore, in modo che si possa dire chi ha fatto cosa. D'altra parte si potrebbe anche volere che nessuno abbia tali privilegi.

1.1.13 Tempo di giocare

Si è ora pronti a giocare con il sistema Debian senza rischi fintanto che si usa l'account utente non privilegiato.

Ciò è possibile perché il sistema Debian è, anche subito dopo l'installazione predefinita, configurato con i permessi dei file corretti che impediscono agli utenti non privilegiati di danneggiare il sistema. Ci possono essere naturalmente ancora dei punti deboli che possono essere sfruttati, ma chi si preoccupa di tali cose non dovrebbe leggere questa sezione ma piuttosto il manuale [Securing Debian](#).

Si può imparare il sistema Debian, come sistema **nix* nelle sezioni seguenti.

- Sezione [1.2](#) (concetti di base)
- Sezione [1.3](#) (sopravvivere)
- Sezione [1.4](#) (lavoro di base)
- Sezione [1.5](#) (uso della shell)
- Sezione [1.6](#) (elaborazione di testi)

1.2 File system stile Unix

In GNU/Linux ed altri sistemi operativi **nix*, i **file** sono organizzati in **directory**. Tutti i file e le directory sono organizzati in un unico grande albero che ha la sua radice in `"/`. Viene chiamato albero perché il file system, se viene disegnato, ha l'aspetto di un albero messo però sottosopra.

Questi file e directory possono essere sparsi su diversi dispositivi. `mount(8)` serve per attaccare il file system trovato su un qualche dispositivo al grande albero dei file. Al contrario, `umount(8)` lo stacca. Nei kernel Linux recenti `mount(8)` con alcune opzioni può collegare parte di un albero dei file in qualche altra posizione o può montare file system come condivisi, privati, slave o non-collegabili. Le opzioni di montaggio supportate per ciascun file system sono disponibili in `"/usr/share/doc/linux-doc-*/Documentation/`

Nei sistemi Unix vengono chiamate **directory** quelle che in altri sistemi sono chiamate **cartelle**. Notare anche che non esiste in nessun sistema Unix un concetto di **unità**, come `"A:"`. C'è un solo unico file system e tutto vi è incluso. Questo è un grandissimo vantaggio rispetto a Windows.

1.2.1 Nozioni di base sui file Unix

Ecco alcune nozioni di base sui file Unix.

- Nei nomi dei file si distingue tra **maiuscole e minuscole**. I file `"MIOFILE"` e `"MioFile"` sono cioè file diversi.
 - La **directory root** indica la radice del file system, indicata semplicemente con `"/`. Non la si confonda con la directory home dell'utente root: `"/root"`.
 - Ogni directory ha un nome che può contenere qualsiasi lettera o simbolo **tranne"/**. La directory radice è un'eccezione; il suo nome è `"/` (pronunciato "slash" o "directory root/radice") e il suo nome non può essere cambiato.
-

- Ogni file o directory è indicato da un **nome di file pienamente qualificato**, un **nome file assoluto** o un **percorso**, fornendo la sequenza di directory attraverso le quali si deve passare per raggiungerlo. I tre termini sono sinonimi.
- Tutti i **nomi di file pienamente qualificati** iniziano con la directory `"/"` e c'è un carattere `"/"` tra ciascuna directory o file nel nome del file. Il primo carattere `"/"` è la directory di più alto livello e gli altri `"/"` separano le directory successive fino a che non si raggiunge l'ultima voce che è il nome del file stesso. I termini usati possono creare confusione. Prendere in considerazione come esempio il seguente **nome file pienamente qualificato** come esempio: `"/usr/share/keytables/us.map.gz"`; anche se ci si riferisce al solo nome base, `"us.map.gz"` come al suo nome file.
- La directory radice ha un certo numero di rami, come `"/etc/"` e `"/usr/"`. Queste sottodirectory a loro volta si diramano in ulteriori sottodirectory, come `"/etc/init.d/"` e `"/usr/local/"`. Viste tutte insieme vengono chiamate **albero delle directory**. Si può pensare ad un nome file assoluto come ad un percorso dalla base dell'albero (`"/"`) alla punta di un ramo (un file). Si può anche sentir parlare dell'albero delle directory come se fosse un albero **genealogico** che comprende tutti i discendenti di un'unica figura detta directory radice (`"/"`): le sottodirectory hanno perciò dei **genitori** e un percorso mostra gli antenati completi di un file. Ci sono anche percorsi relativi che iniziano da una qualche posizione che non sia la directory radice. Si dovrebbe tenere a mente che la directory `"/../"` si riferisce alla directory genitore. Questa terminologia si applica anche ad altre strutture simili a quella delle directory, come le strutture di dati gerarchici.
- Non c'è alcun nome speciale di percorso di directory che corrisponda ad un dispositivo fisico, come il disco fisso. Questa è una differenza rispetto a [RT-11](#), [CP/M](#), [OpenVMS](#), [MS-DOS](#), [AmigaOS](#) e [Microsoft Windows](#), in cui il percorso contiene una porzione con il nome di dispositivo come `"C:\\"`. (Esistono tuttavia directory che si riferiscono ai dispositivi fisici come a parte del file system normale. Vedere Sezione [1.2.2](#).)

Nota

Benché si **possa** usare quasi qualsiasi lettera o simbolo nel nome di un file, in pratica farlo non è una bella idea. È meglio evitare qualsiasi carattere che ha spesso un significato particolare sulla riga di comando, inclusi spazi, tabulazioni, a capo e altri caratteri speciali: `{ } () [] ' ` " \ / > < | ; ! # & ^ * % @ $`. Se si vuole separare delle parole all'interno di un nome file, buone alternative sono un punto, un trattino ed il segno di sottolineatura. Si può anche scrivere in maiuscolo ogni parola, `"InQuestoModo"`. Gli utenti Linux esperti tendono ad evitare l'uso degli spazi nei nomi dei file.

Nota

La parola `"root"` può significare l'"utente root" o la "directory root". Il contesto in cui il termine viene usato dovrebbe rendere chiaro il suo significato.

Nota

La parola **percorso** non è usata solamente per un **nome di file pienamente qualificato** come descritto in precedenza, ma anche per il **percorso di ricerca dei comandi**. Il significato è di solito reso chiaro dal contesto.

Le linee guida raccomandate per la gerarchia dei file sono descritte in dettaglio nel Filesystem Hierarchy Standard (`"/usr/share/doc/` e in `hier(7)`). Come inizio si dovrebbe tenere a mente quanto segue.

directory	uso della directory
<code>/</code>	la directory root
<code>/etc/</code>	file di configurazione a livello di sistema
<code>/var/log/</code>	file di registro del sistema
<code>/home/</code>	tutte le directory home degli utenti non privilegiati

Tabella 1.3: Elenco degli usi delle directory principali

1.2.2 Aspetti tecnici del file system

Nella scia della **tradizione Unix**, il sistema Debian GNU/Linux fornisce il **file system** in cui risiedono i dati fisici nel disco fisso e negli altri dispositivi di memorizzazione, ed inoltre le interazioni con i dispositivi hardware come gli schermi delle console e le console seriali remote vengono rappresentate in un modo unificato in `"/dev/"`.

Ogni file, directory, pipe con nome (un modo per due programmi di scambiare dati) o dispositivo fisico presente in un sistema Debian GNU/Linux ha una struttura di dati chiamata **inode** che descrive gli attributi ad esso associati come l'utente che lo possiede (proprietario), il gruppo a cui appartiene, la data dell'ultimo accesso ad esso, ecc. L'idea di rappresentare praticamente tutto nel file system è stata un'innovazione di Unix e i kernel Linux moderni hanno sviluppato questa idea e sono andati oltre. Ora anche le informazioni sui processi in esecuzione sul computer possono essere trovate nel file system.

La rappresentazione astratta e unificata di entità fisiche e di processi interni è molto potente dato che permette di usare lo stesso comando per lo stesso tipo di operazione su molti tipi di device completamente diversi l'uno dall'altro. È anche possibile cambiare il modo in cui il kernel funziona scrivendo dati in file speciali che sono collegati ai processi in esecuzione.

Suggerimento

Per identificare la corrispondenza tra l'albero dei file e le entità fisiche, eseguire `mount(8)` senza opzioni.

1.2.3 Permessi del file system

I **permessi del file system** di sistemi ***nix** sono definiti e influenzano tre categorie di utenti.

- L'**utente** che è il proprietario del file (**u**).
- Gli altri utenti del **gruppo** a cui appartiene il file (**g**).
- Tutti gli **altri** utenti (**o**) a cui ci si riferisce anche con i termini "mondo" e "tutti".

Per un file, a ciascun permesso corrispondono le seguenti azioni.

- Il permesso di **lettura** (**r**) permette al proprietario di esaminare il contenuto del file.
- Il permesso di **scrittura** (**w**) permette al proprietario di modificare il file.
- Il permesso di **esecuzione** (**x**) permette al proprietario di eseguire il file come comando.

Per una directory, a ciascun permesso corrispondono le seguenti azioni.

- Il permesso di **lettura** (**r**) permette al proprietario di elencare il contenuto della directory.
- Il permesso di **scrittura** (**w**) permette al proprietario di aggiungere o rimuovere file dalla directory.
- Il permesso di **esecuzione** (**x**) permette al proprietario di accedere ai file nella directory.

In questo caso il permesso di **esecuzione** su una directory non solo significa poter leggere i file in quella directory ma anche poterne vedere gli attributi, come la dimensione e la data di modifica.

Per visualizzare le informazioni sui permessi (ed altro) di file e directory si usa `ls(1)`. Quando chiamato con l'opzione `"-l"` mostra, nell'ordine elencato in seguito, le seguenti informazioni.

- **Tipo di file** (primo carattere).
 - **Permessi** di accesso al file (nove caratteri, tre ciascuno per utente, gruppo e altri, in questo ordine).
 - **Numero di collegamenti fisici** al file.
 - Nome dell'**utente** proprietario del file.
-

carattere	significato
-	file normale
d	directory
l	collegamento simbolico
c	device a caratteri
b	device a blocchi
p	pipe con nome
s	socket

Tabella 1.4: Elenco dei valori possibili per il primo carattere nell'output di "ls -l"

- Nome del **gruppo** a cui il file appartiene.
- **Dimensione** in caratteri (byte) del file.
- **Data ed ora** del file (mtime).
- **Nome** del file.

Per cambiare il proprietario di un file da un account root si usa `chown(1)`. Per cambiare il gruppo di un file dall'account del proprietario o da quello di root si usa `chgrp(1)`. Per cambiare i permessi di accesso di file o directory dall'account del proprietario o da quello di root si usa `chmod(1)`. La sintassi di base per manipolare un file `pippo` è la seguente.

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

Per esempio si può fare in modo che l'utente `pippo` sia il proprietario di un albero di directory condivisa dal gruppo `pluto` con i seguenti comandi.

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

Ci sono altri tre bit speciali di permessi.

- Il bit **set user ID** (`s` o `S` al posto del permesso `x` dell'utente).
- Il bit **set group ID** (`s` o `S` al posto del permesso `x` del gruppo).
- Il bit **sticky** (`t` o `T` al posto del permesso `x` degli altri).

Nell'output di "ls -l" il valore di questi bit è **maiuscolo** se il bit di permesso di esecuzione nascosto da essi è **non impostato**.

L'impostazione del bit **set user ID** per un file eseguibile permette ad un utente di eseguire quel file con l'ID del proprietario del file (per esempio `root`). In modo simile, l'impostazione del bit **set group ID** per un file eseguibile permette ad un utente di eseguire il file con l'ID del gruppo a cui appartiene il file (per esempio `root`). Poiché l'impostazione di questi bit può causare rischi in termini di sicurezza, il loro uso richiede una particolare cautela.

L'impostazione del bit **set group ID** per una directory abilita lo schema di creazione di file in [stile BSD](#), in cui tutti i file creati nella directory appartengono al **gruppo** della directory.

L'impostazione dello **sticky bit** per una directory impedisce la rimozione di un file nella directory da parte dell'utente che non ne è il proprietario. Per preservare i contenuti di un file in directory con permessi di scrittura per tutti, come `/tmp` o in directory con permessi di scrittura per un gruppo, non solo si deve impostare il permesso di **scrittura** per il file, ma anche impostare lo **sticky bit** per la directory. In caso contrario, qualunque utente con i permessi di scrittura per la directory può rimuovere il file e crearne uno nuovo con lo stesso nome.

Ecco alcuni interessanti esempi di permessi dei file.

```
$ 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
```

Esiste un metodo numerico alternativo per descrivere i permessi dei file con `chmod(1)`; tale metodo numerico usa numeri ottali (base=8) di 3 o 4 cifre.

cifra	significato
1 ^a cifra opzionale	somma di set user ID (=4), set group ID (=2) e sticky bit (=1)
2 ^a cifra	somma dei permessi di lettura (=4), scrittura (=2) e esecuzione (=1) per l' utente
3 ^a cifra	come sopra, ma per il gruppo
4 ^a cifra	come sopra, ma per gli altri

Tabella 1.5: Permessi in notazione numerica per i comandi `chmod(1)`

Sembra complicato, ma è in realtà piuttosto semplice. Se si guardano le prime colonne (2-10) nell'output del comando `ls -l` e le si leggono come una rappresentazione binaria (base=2) dei permessi sui file (dove `-` equivale a `0` e `rwX` equivalgono a `1`), le ultime 3 cifre del valore numerico dei permessi dovrebbero apparire come la corretta rappresentazione dei permessi sui file in numerazione ottale (base=8).

Per esempio, provare a fare quanto segue.

```
$ 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
```

Suggerimento

Se si deve aver accesso alle informazioni visualizzate da `ls -l` in script di shell, si dovrebbero usare i comandi pertinenti, come `test(1)`, `stat(1)` e `readlink(1)`. Si possono usare anche i comandi interni della shell come `["` o `test`.

1.2.4 Controllo dei permessi per i file appena creati: `umask`

I permessi che vengono applicati ad un file o una directory appena creati sono limitati dal comando interno della shell `umask`. Vedere `dash(1)`, `bash(1)` e `builtins(7)`.

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

Il sistema Debian usa, in modo predefinito, uno schema UPG (User Private Group, gruppo privato dell'utente). Ogni volta che viene aggiunto un nuovo utente al sistema, viene creato un UPG; questo ha lo stesso nome dell'utente per il quale è stato creato e quell'utente è l'unico membro dell'UPG. Lo schema UPG rende sicura l'impostazione della `umask` a `0002` dato che ogni utente ha il proprio gruppo privato. (In alcune varianti di Unix è abbastanza comune impostare tutti gli utenti normali come appartenenti ad un unico gruppo **users** ed è in quel caso una buona idea impostare `umask` a `0022` per ragioni di sicurezza.)

umask	permessi dei file creati	permessi delle directory create	uso
0022	-rw-r--r--	-rwxr-xr-x	scrivibile solo dall'utente
0002	-rw-rw-r--	-rwxrwxr-x	scrivibile solo dal gruppo

Tabella 1.6: Esempi di valori di **umask****Suggerimento**

Abilitare UPG inserendo "umask 002" nel file ~/.bashrc.

1.2.5 Permessi per gruppi di utenti (gruppi)

Per far sì che i permessi di un gruppo vengano applicati ad un particolare utente, tale utente deve essere inserito come membro del gruppo usando "sudo vigr" per /etc/group e "sudo vigr -s" per /etc/gshadow. Per abilitare la nuova configurazione dei gruppi è necessario fare il log out e quindi di nuovo il login (oppure eseguire "exec newgrp").

Nota

In alternativa, si possono aggiungere dinamicamente gli utenti ai gruppi durante il processo di autenticazione aggiungendo la riga "auth optional pam_group.so" al file "/etc/pam.d/common-auth" e configurando "/etc/security/group.conf". (Vedere Capitolo 4.)

I dispositivi hardware sono, in un sistema Debian, semplicemente un altro tipo di file. Se si hanno problemi ad accedere a dispositivi quali CD-ROM e chiavette USB da un account utente, si dovrebbe inserire quell'utente nel gruppo appropriato.

Alcuni importanti gruppi pre-impostati dal sistema permettono ai loro membri l'accesso a file e device particolari senza i privilegi di root.

gruppo	descrizione dei file e device accessibili
di <code>al</code> out	accesso diretto e completo alle porte seriali ("/dev/ttyS[0-3]")
dip	accesso limitato alle porte seriali per connessione IP dialup a peer fidati
cdrom	unità CD-ROM, DVD+/-RW
audio	device audio
video	device video
scanner	scanner
adm	registri di monitoraggio del sistema
staff	alcune directory per compiti minori di amministrazione: ""/usr/local", "/home"

Tabella 1.7: Elenco dei principali gruppi forniti dal sistema per accesso ai file

Suggerimento

È necessario far parte del gruppo dialout per riconfigurare il modem, comporre numeri, ecc. Se però root crea file di configurazione predefiniti per i peer fidati in "/etc/ppp/peers/", è necessario solamente far parte del gruppo dip per creare una connessione **IP dialup** a tali peer fidati usando i comandi pppd(8), pon(1) e poff(1).

Alcuni importanti gruppi pre-impostati dal sistema permettono ai loro membri di eseguire particolari comandi senza i privilegi di root.

Per l'elenco completo degli utenti e gruppi forniti dal sistema, vedere la recente versione del documento "Utenti e gruppi" in "/usr/share/doc/base-passwd/users-and-groups.html" fornito dal pacchetto base-passwd.

Vedere passwd(5), group(5), shadow(5), newgrp(1), vipw(8), vigr(8) e pam_group(8) per informazioni sui comandi di gestione di utenti e gruppi sul sistema.

gruppo	comandi accessibili
sudo	eseguire <code>SUDO</code> senza password
lpadmin	eseguire comandi per aggiungere, modificare e rimuovere stampanti dal database delle stampanti

Tabella 1.8: Elenco dei principali gruppi forniti dal sistema per l'esecuzione di particolari comandi

1.2.6 Orari

Ci sono tre tipi di orari per un file GNU/Linux.

tipo	significato (definizione storica Unix)
mtime	orario di modifica del file (<code>ls -l</code>)
ctime	orario di cambiamento di stato del file (<code>ls -lc</code>)
atime	orario dell'ultimo accesso al file (<code>ls -lu</code>)

Tabella 1.9: Elenco dei tipi di data

Nota

ctime non è l'orario di creazione del file.

Nota

L'attuale significato di **atime** sui sistemi GNU/Linux potrebbe essere diverso dalla sua definizione storica in Unix.

- La sovrascrittura di un file cambia tutti gli attributi **mtime**, **ctime** e **atime** del file.
- Il cambiamento del proprietario o dei permessi di un file cambia gli attributi **ctime** e **atime** del file.
- La lettura di un file cambia l'attributo **atime** di un file nei sistemi Unix storici.
- La lettura di un file cambia l'attributo **atime** di un file in un sistema GNU/Linux se il file system è montato con "strictatime".
- Leggere un file per la prima volta o dopo un giorno modifica l'attributo **atime** del file stesso su sistemi GNU/Linux, se il filesystem è stato montato con "relatime". (comportamento di default da Linux 2.6.30)
- Leggere un file non modifica l'attributo **atime** del file stesso su sistemi GNU/Linux se il filesystem è stato montato con "noatime".

Nota

Le opzioni di mount "noatime" e "relatime" sono introdotte per migliorare le prestazioni in lettura del file system nei normali casi d'uso. Semplici operazioni di lettura dei file con l'opzione "strictatime" comportano lunghe operazioni di scrittura per aggiornare l'attributo **atime**. L'attributo **atime** però è raramente utilizzato, a parte per il file `mbox(5)`. Vedere `mount(8)`.

Usare il comando `touch(1)` per cambiare l'orario di file esistenti.

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
```

Suggerimento

Vedere Sezione [9.3.4](#) per personalizzare l'output di "ls -l".

1.2.7 Collegamenti

Ci sono due metodi per associare un file "pippo" ad un diverso nome file "pluto".

- [Collegamento fisico](#)
 - Nome duplicato per un file esistente
 - "ln pippo pluto"
- [Collegamento simbolico o symlink](#)
 - File speciale che punta ad un altro file in base al nome
 - "ln -s pippo pluto"

Vedere l'esempio seguente per notare i cambiamenti nel conteggio dei collegamenti e le sottili differenze tra i risultati del 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.

Il collegamento simbolico ha sempre permessi di accesso nominali "rwxrwxrwx", come mostrato nell'esempio precedente, ma con gli effettivi permessi di accesso stabiliti dai permessi del file a cui punta.

**Attenzione**

In generale è una buona idea non creare collegamenti simbolici complicati o non creare collegamenti fisici per nulla, a meno di non avere una ragione molto buona per farlo. Possono diventare degli incubi quando la combinazione logica dei collegamenti simbolici crea cicli ricorsivi nel file system.

Nota

È in generale preferibile usare collegamenti simbolici piuttosto che fisici, a meno che non sia abbia una buona ragione per usare un collegamento fisico.

La directory "." punta alla directory in cui appare, perciò il conteggio dei collegamenti per ogni nuova directory inizia da 2. La directory ".." punta alla directory genitore, perciò il conteggio dei collegamenti di una directory aumenta con l'aggiunta di nuove sottodirectory.

Se si è appena passati da Windows a Linux, appare presto chiaro come sia ben progettato il collegamento dei nomi di file in Unix se paragonato con i collegamenti in ambiente Windows che sono l'equivalente più prossimo in quel sistema. Dato che sono implementati nel file system, le applicazioni non possono vedere alcuna differenza tra un file collegamento ed un originale. Nel caso di collegamenti fisici, non c'è realmente nessuna differenza.

1.2.8 Pipe con nome (FIFO)

Una [pipe con nome](#) è un file che si comporta da pipe. Si mette qualcosa dentro il file e questa cosa esce dall'altra parte. Questa è chiamata una FIFO (First-In-First-Out, primo ad entrare-primo ad uscire): la prima cosa che viene immessa nella pipe è la prima ad uscire dall'altra estremità.

Se si scrive su una pipe con nome, il processo che sta eseguendo la scrittura nella pipe non termina fino a che l'informazione scritta nella pipe non viene letta. Se si legge da una pipe con nome, il processo di lettura attende che non ci sia più nulla da leggere prima di terminare. La dimensione della pipe è sempre zero: non archivia dati, ma semplicemente collega due processi come fa la funzionalità fornita dalla sintassi per pipe "|" della shell. Tuttavia, dato che questa pipe ha un nome, i due processi non devono essere nella stessa riga di comando e nemmeno essere eseguiti dallo stesso utente. Le pipe sono un'innovazione di Unix di grandissimo impatto.

Per esempio, provare a fare quanto segue.

```
$ 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 Socket

I socket sono usati moltissimo da tutte le comunicazioni Internet, da database e dal sistema operativo stesso. Sono simili alle pipe con nome (FIFO) e permettono ai processi di scambiare informazioni anche tra computer diversi. Per un socket, non è necessario che questi processi siano in esecuzione contemporaneamente, né di essere eseguiti come processi figli di uno stesso processo antenato. Questo è il punto culminante della [comunicazione tra processi \(IPC\)](#). Lo scambio di informazioni può avvenire sulla rete tra host diversi. I due più comuni sono il [socket Internet](#) e gli [Unix domain socket](#).

Suggerimento

"netstat -an" fornisce una vista d'insieme molto utile dei socket aperti su un dato sistema.

1.2.10 File di device

I [file di device](#) fanno riferimento a dispositivi fisici o virtuali sul sistema, come i dischi fissi, la scheda video, lo schermo o la tastiera. Un esempio di dispositivo virtuale è la console, rappresentata da "/dev/console".

Ci sono 2 tipi di file di device.

- **Device a caratteri**
-

- Vi si accede un carattere alla volta.
- 1 carattere = 1 byte
- Es., device della tastiera, porta seriale, ...

- **Device a blocchi**

- Vi si accede in base a unità più grandi chiamate blocchi.
- 1 blocco > 1 byte
- Es., dischi fissi, ...

I file di device possono essere letti e scritti, anche se è probabile che i file contengano dati binari che appaiono come farfugli incomprensibili per le persone. La scrittura diretta di dati in questi file è utile a volte per trovare la soluzione a problemi con le connessioni hardware. Si può, per esempio, fare il dump di un file di testo in un device di stampa `"/dev/lp0"` o inviare comandi del modem alla porta seriale appropriata `"/dev/ttyS0"`. Ma, a meno che ciò non venga fatto con cautela, può causare grandissimi disastri. Perciò, essere prudenti.

Nota

Per il normale accesso ad una stampante, usare `lp(1)`.

I numeri di nodo dei device sono visualizzati se si esegue `ls(1)` come nell'esempio seguente.

```
$ 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"` ha major number del device 8 e minor number del device 0. È accessibile in lettura e scrittura dagli utenti che appartengono al gruppo `disk`.
- `"/dev/sr0"` ha major number del device 11 e minor number del device 0. È accessibile in lettura e scrittura dagli utenti che appartengono al gruppo `cdrom`.
- `"/dev/ttyS0"` ha major number del device 4 e minor number del device 64. È accessibile in lettura e scrittura dagli utenti che appartengono al gruppo `dialout`.
- `"/dev/zero"` ha major number del device 1 e minor number del device 5. È accessibile in lettura e scrittura da chiunque.

Nei sistemi Linux moderni il file system sotto `"/dev"` viene popolato automaticamente dal meccanismo `udev(7)`.

1.2.11 File di device speciali

Ci sono alcuni file di device speciali.

Sono spesso usati insieme alla redirectione di shell (vedere Sezione [1.5.8](#)).

1.2.12 `procfs` e `sysfs`

`procfs` e `sysfs`, montati in `"/proc"` e `"/sys"` sono pseudo-file system ed espongono strutture interne di dati del kernel nello spazio utente. In altre parole, queste voci sono virtuali, funzionano cioè come una comoda finestra sul funzionamento del sistema operativo.

La directory `"/proc"` contiene (tra le altre cose) una sottodirectory per ciascun processo in esecuzione sul sistema che prende nome dall'ID del processo (PID). Le utilità di sistema, come `ps(1)`, che accedono alle informazioni sui processi, ottengono le loro informazioni da questa struttura di directory.

file di device	azione	descrizione della risposta
/dev/null	lettura	restituisce il "carattere EOF (End of File, fine del file)"
/dev/null	scrittura	non ritorna nulla (un pozzo senza fondo in cui buttare via dati)
/dev/zero	lettura	ritorna il "carattere \0 (NUL)", diverso dal numero zero in ASCII
/dev/random	lettura	ritorna caratteri a caso da un vero generatore di numeri casuali che dà reale entropia (lento)
/dev/urandom	lettura	ritorna caratteri casuali da un generatore di numeri pseudocasuali crittograficamente sicuro
/dev/full	scrittura	ritorna l'errore di disco pieno (ENOSPC)

Tabella 1.10: Elenco di file dei device speciali

Le directory in `"/proc/sys/"` contengono interfacce per cambiare certi parametri del kernel durante l'esecuzione. (Si può fare la stessa cosa tramite comandi `sysctl(8)` specifici o tramite il suo file di precaricamento/configurazione `"/etc/sysctl.conf"`.)

Le persone spesso vanno in panico quando si accorgono di un file in particolare, `"/proc/kcore"` che è particolarmente enorme. È (più o meno) una copia del contenuto della memoria del computer ed è usato per fare il debug del kernel. È un file virtuale che punta alla memoria del computer perciò non ci si preoccupi della sua dimensione.

La directory `"/sys"` contiene strutture dati del kernel esportate, i loro attributi e i collegamenti tra di esse. Contiene anche interfacce per cambiare alcuni parametri del kernel durante l'esecuzione.

Vedere `proc.txt(.gz)`, `sysfs.txt(.gz)` e altri documenti correlati nella documentazione del kernel Linux (`"/usr/share/"` fornita dai pacchetti `linux-doc-*`).

1.2.13 tmpfs

`tmpfs` è un file system temporaneo che contiene tutti i file nella [memoria virtuale](#). I dati del `tmpfs` nella [page cache](#) in memoria possono essere spostati nello [spazio di swap](#) su disco quando necessario.

La directory `"/run"` viene montata come il `tmpfs` nelle prime fasi del processo di avvio. Ciò vi permette la scrittura anche quando la directory `"/"` è montata in sola lettura. Questa è la nuova posizione per la memorizzazione dei file transitori e sostituisce diverse posizioni descritte nella versione 2.3 del [Filesystem Hierarchy Standard](#) (Standard per la gerarchia del file system):

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

Vedere `tmpfs.txt(.gz)` nella documentazione del kernel Linux (`"/usr/share/doc/linux-doc-*/Documentation/"` fornita dai pacchetti `linux-doc-*`).

1.3 Midnight Commander (MC)

[Midnight Commander \(MC\)](#) è un "coltellino svizzero" GNU per la console Linux ed altri ambienti in terminale. Dà ai principianti la possibilità di usare una console comandata da menu che è molto più facile da imparare dei comandi Unix standard.

Potrebbe essere necessario installare il pacchetto di Midnight Commander che si chiama `"mc"` con il seguente comando.

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

Usare il comando `mc(1)` per esplorare il sistema Debian. È il modo migliore di imparare. Esplorare qualche posizione interessante usando semplicemente i tasti freccia e il tasto Invio.

- `"/etc"` e le sue sottodirectory
- `"/var/log"` e le sue sottodirectory
- `"/usr/share/doc"` e le sue sottodirectory
- `"/sbin"` e `"/bin"`

1.3.1 Personalizzazione di MC

Per far sì che MC cambi la directory di lavoro in uscita e si sposti con `cd` nella directory, suggerisco di modificare `~/.bashrc` per includere uno script fornito nel pacchetto `mc`.

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

Vedere `mc(1)` (sotto l'opzione `"-P"`) per capirne la ragione. (Se non è chiaro esattamente quello di cui sto parlando, si può tornare a questo più tardi.)

1.3.2 Avvio di MC

MC può essere avviato nel modo seguente.

```
$ mc
```

MC si prende cura di tutte le operazioni sui file attraverso i suoi menu, richiedendo perciò solo un minimo sforzo da parte dell'utente. Premere semplicemente `F1` per ottenere la schermata di aiuto. Si può giocherellare con MC premendo semplicemente i tasti freccia e i tasti funzione.

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 si incontrano problemi di codifica dei caratteri con visualizzazione di caratteri spazzatura, l'aggiunta dell'opzione `"-a"` alla riga di comando di MC può aiutare a prevenirli.

Se ciò non risolve i problemi di visualizzazione in MC, vedere Sezione [9.5.6](#).

1.3.3 Gestore dei file in MC

L'impostazione predefinita è con due pannelli di directory contenenti gli elenchi dei file. Un'altra utile modalità è l'impostazione della finestra destra per contenere le "informazioni" per vedere informazioni sui privilegi di accesso dei file, ecc. Di seguito sono elencati alcuni tasti essenziali. Con il demone `gpm(8)` in esecuzione, si può anche usare il mouse in una console a caratteri Linux. (Assicurarsi di premere il tasto Maiusc per ottenere il comportamento normale di taglia e incolla in MC.)

1.3.4 Trucchetti per la riga di comando di MC

- Il comando `cd` cambia la directory mostrata nel pannello selezionato.
 - `Ctrl-Invio` o `Alt-Invio` copia un nome di file nella riga di comando. Usarlo con i comandi `cp(1)` e `mv(1)` assieme alla modifica della riga di comando.
 - `Alt-Tab` mostra le scelte date dall'espansione di shell dei nomi di file.
 - Si possono specificare le directory iniziali per entrambe le finestre come argomenti per MC; per esempio, `"mc /etc /root"`.
 - `Esc + tasto n` → `Fn` (cioè, `ESC + 1` → `F1`, ecc.; `ESC + 0` → `F10`)
 - Premere `ESC` prima di un tasto ha lo stesso effetto di premere `Alt` ed il tasto contemporaneamente; premere cioè `ESC + c` per `Alt-C`. `ESC` è chiamato meta-tasto ed è a volte indicato con `"M-"`.
-

tasto	azione associata
F1	menu di aiuto
F3	visualizzatore interno di file
F4	editor interno
F9	attiva il menu a tendina
F10	esce da Midnight Commander
Tab	sposta tra le due finestre
Insert o Ctrl-T	segna il file per una operazione su più file, come la copia
Del	cancella il file (essere prudenti: impostare MC per la modalità di cancellazione sicura)
tasti freccia	intuitivi

Tabella 1.11: Le associazioni dei tasti di MC

1.3.5 L'editor interno di MC

L'editor interno ha un interessante schema per il taglia e incolla. La pressione di F3 marca l'inizio di una selezione, una seconda pressione di F3 marca la fine della selezione e la evidenzia. Si può poi muovere il cursore. Se si preme F6, l'area selezionata viene spostata nella posizione del cursore. Se si preme F5 l'area selezionata viene copiata ed inserita alla posizione del cursore. F2 salva il file. F10 fa uscire. La maggior parte dei tasti cursore funziona in modo intuitivo.

Si può avviare direttamente questo editor su di un file usando uno dei comandi seguenti.

```
$ mc -e filename_to_edit
```

```
$ mcedit filename_to_edit
```

Questo non è un editor multi-finestra, ma si possono usare più console Linux per ottenere lo stesso effetto. Per copiare tra finestre, usare i tasti `Alt-Fn` per cambiare console virtuale e usare "File → Insert file" o "File → Copy to file" per spostare una porzione di un file in un altro file.

Questo editor interno può essere rimpiazzato da qualsiasi editor esterno a propria scelta.

Inoltre molti programmi usano la variabile d'ambiente "\$EDITOR" o "\$VISUAL" per decidere quale editor usare. Se inizialmente non ci si trova a proprio agio con `vim(1)` o `nano(1)`, si può impostare queste variabili a "mcedit" aggiungendo le righe seguenti al file "~/.bashrc".

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

Io raccomando di impostarle a "vim", se possibile.

Se non ci si trova a proprio agio con `vim(1)`, si può continuare ad usare `mcedit(1)` per la maggior parte dei compiti di manutenzione del sistema.

1.3.6 Il visualizzatore interno di MC

MC è un visualizzatore molto intelligente. È un grande strumento per cercare parole in documenti. Lo uso sempre per i file nella directory `"/usr/share/doc"`. È il metodo più veloce per navigare tra una massa di informazioni Linux. Questo visualizzatore può essere avviato direttamente usando uno dei comandi seguenti.

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

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

tipo di file	reazione al tasto Invio
file eseguibile	esegue comando
file man	invia i contenuti tramite pipe al software di visualizzazione
file html	invia i contenuti tramite pipe al browser web
file <code>**.tar.gz</code> e <code>**.deb</code>	naviga i suoi contenuti come se fosse una sottodirectory

Tabella 1.12: La reazione al tasto Invio in MC

1.3.7 Funzionalità di avvio automatico di MC

Se si preme Invio su di un file, un programma appropriato gestirà il contenuto del file (vedere Sezione 9.4.11). Questa è una funzionalità molto comoda di MC.

Per far sì che queste funzionalità di visualizzazione dei file e di file virtuali funzionino, i file visualizzabili non dovrebbero essere impostati come eseguibili. Cambiare il loro stato usando `chmod(1)` o attraverso il menu file di MC.

1.3.8 File system FTP virtuale di MC

MC può essere usato per accedere a file in Internet usando FTP. Premere F9 per andare nel menu, poi digitare "p" per attivare il file system FTP virtuale. Inserire un URL nella forma `nomeutente:password@nomehost.nomedominio` e la directory remota apparirà come una directory locale.

Provare `[deb.debian.org/debian]` come URL e navigare nell'archivio Debian.

1.4 Ambiente di lavoro di base in stile Unix

Anche se MC permette di fare quasi tutto, è bene imparare come usare gli strumenti a riga di comando invocati dal prompt di shell e prendere familiarità con l'ambiente di lavoro in stile Unix.

1.4.1 La shell di login

Si può selezionare la propria shell di login con `chsh(1)`.

pacchetto	popcon	dimensione	Shell POSIX	descrizione
bash	V:796, I:999	6470	Sì	Bash : GNU Bourne Again SHell (standard de facto)
bash-completion	V:31, I:922	1523	N/D	programmabile completion for the bash shell
dash	V:913, I:993	221	Sì	Debian Almquist Shell , buona per script di shell
zsh	V:36, I:74	2467	Sì	Z shell : la shell standard con molti miglioramenti
tcsh	V:8, I:28	1316	No	TENEX C Shell : una versione migliorata di Berkeley csh
mksh	V:7, I:12	1478	Sì	Una versione della Korn shell
csh	V:2, I:8	343	No	C Shell OpenBSD , una versione di Berkeley csh
sash	V:0, I:6	1090	Sì	Stand-alone shell con comandi interni (non adatta per essere la <code>"/bin/sh"</code> standard)
ksh	V:2, I:15	3284	Sì	la versione reale di AT&T della Korn shell
rc	V:0, I:1	169	No	implementazione della rc shell di AT&T Plan 9
posh	V:0, I:0	190	Sì	Policy-compliant Ordinary SHell (derivata da <code>pdksh</code>)

Tabella 1.13: Elenco di programmi shell

Suggerimento

Sebbene le shell in stile POSIX condividano la stessa sintassi di base, esse possono differire nel comportamento relativo a cose anche basilari, come le variabili della shell e l'espansione dei modelli glob. Per i dettagli controllare la loro documentazione.

In questo capitolo del tutorial la shell interattiva considerata è sempre `bash`.

1.4.2 Personalizzare bash

Si può personalizzare il comportamento di `bash(1)` con `~/ .bashrc`.

Per esempio provare quanto segue.

```
# 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
```

Suggerimento

Si possono trovare altri suggerimenti sulla personalizzazione di `bash(1)`, come Sezione [9.3.6](#), in Capitolo [9](#).

Suggerimento

Il pacchetto `bash-completion` permette il completamento automatico per `bash`.

1.4.3 Associazioni di tasti speciali

Nell'ambiente **nix*, ci sono alcune associazioni di tasti che hanno un significato speciale. Notare che in una console a caratteri Linux normale solo i tasti `Ctrl` e `Alt` sinistri funzionano come atteso. Ecco alcune combinazioni di tasti che vale la pena ricordare.

Suggerimento

La funzionalità di `Ctrl-S` nel terminale può essere disabilitata usando `stty(1)`.

tasto	descrizione dell'associazione di tasti
Ctrl-U	cancella il contenuto della riga prima del cursore
Ctrl-H	cancella il carattere prima del cursore
Ctrl-D	termina l'input (se si sta usando la shell, esce dalla shell)
Ctrl-C	termina un programma in esecuzione
Ctrl-Z	arresta temporaneamente un programma spostandolo come compito sullo sfondo
Ctrl-S	ferma l'output a schermo.
Ctrl-Q	riattiva l'output a schermo.
Ctrl-Alt-Del	riavvia/ferma il sistema, vedere <code>inittab(5)</code>
Left-Alt-key (opzionalmente, tasto Windows)	meta-tasto per Emacs e interfacce utente simili
freccia in su	avvia la ricerca nello storico dei comandi in <code>bash</code>
Ctrl-R	inizia una ricerca incrementale nello storico dei comandi in <code>bash</code>
Tab	completa l'input di un nome di file nella riga di comando in <code>bash</code>
Ctrl-V Tab	inserisce in input un carattere <code>Tab</code> senza espansione nella riga di comando in <code>bash</code>

Tabella 1.14: Elenco di associazioni di tasti per `bash`

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`

azione	risposta
Clic sinistro e trascinamento	select range as PRIMARY selection
Clic sinistro	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

Tabella 1.15: List of mouse operations and related key actions on Debian

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.

1.4.5 Il paginatore

Il comando `less(1)` è il paginatore (navigatore del contenuto dei file) migliorato. Legge il file specificato nell'argomento del comando o il suo standard input. Premere "h" se si ha bisogno di aiuto nella navigazione con il comando `less`. Può fare molto di più di `more(1)` e può essere fornito di superpoteri eseguendo `eval $(lesspipe)` o `eval $(lessfile)` nello script di avvio della shell. Vedere ulteriori informazioni in `/usr/share/doc/less/LESSOPEN`. L'opzione "-R" permette output raw e abilita le sequenze di escape ANSI per i colori. Vedere `less(1)`.

1.4.6 L'editor di testo

Si dovrebbe diventare competenti in una delle varianti dei programmi [Vim](#) o [Emacs](#) che sono popolari sui sistemi *nix.

Penso che abituarsi ai comandi Vim sia la cosa giusta da fare, dato che un editor Vi è sempre presente nel mondo Linux/Unix. (In realtà, il `vi` originale o il nuovo `nvi` sono programmi che si trovano ovunque. Io ho scelto invece Vim per i principianti dato che offre l'aiuto attraverso il tasto F1 pur essendo abbastanza simile e più potente.)

Se si sceglie invece [Emacs](#) o [XEmacs](#) come editor preferito, si sta facendo comunque davvero un'ottima scelta, specialmente per la programmazione. Emacs ha una vastità di altre funzionalità, incluso il funzionamento come newsreader, editor di directory, programma di posta, ecc. Quando usato per la programmazione o la scrittura di script, riconosce intelligentemente il formato che si sta usando e cerca di fornire assistenza. Alcune persone sostengono che l'unico programma di cui hanno bisogno su Linux è Emacs. Dieci minuti spesi ad imparare Emacs ora possono far risparmiare ore in futuro. È caldamente raccomandato avere a portata di mano il manuale di GNU Emacs da consultare quando si sta imparando Emacs.

Tutti questi programmi sono di solito forniti con un programma tutor che aiuta ad impararli facendo pratica. Avviare Vim digitando "vim" e premere il tasto F1. Si dovrebbero leggere come minimo le prime 35 righe. Poi seguire il corso di apprendimento in linea muovendo il cursore su "| tutor |" e premendo `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 Impostare un editor di testi predefinito

Debian è fornita con svariati editor differenti. Si raccomanda di installare, come detto in precedenza, il pacchetto `vim`.

Debian fornisce un accesso unificato all'editor predefinito di sistema attraverso il comando `/usr/bin/editor` in modo che altri programmi (ad esempio `reportbug(1)`) possano richiamarlo. Lo si può modificare con il comando seguente.

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

Io raccomando ai principianti la scelta di `/usr/bin/vim.basic` invece di `/usr/bin/vim.tiny` dato che supporta l'evidenziazione della sintassi.

Suggerimento

Molti programmi usano la variabile d'ambiente "\$EDITOR" o "\$VISUAL" per decidere quale editor usare (vedere Sezione 1.3.5 e Sezione 9.4.11. Per coerenza, in un sistema Debian, impostare queste variabili a `/usr/bin/editor`. (Storicamente "\$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

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

mode	key strokes	azione
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

Tabella 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.

Suggerimento

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 Sezione 9.2.

1.4.9 Registrare le attività della shell

L'output dei comandi di shell può scorrere fuori dallo schermo e potrebbe essere perduto per sempre. È buona norma registrare le attività di shell in un file in modo da poterle riguardare in seguito. Questo tipo di registro è essenziale quando si fa una qualsiasi attività di amministrazione del sistema.

Suggerimento

The new Vim (version >= 8.2) can be used to record the shell activities cleanly using TERMINAL - JOB-mode. See Sezione 1.4.8.

Il metodo base per registrare l'attività di shell è di eseguirla sotto script(1).

Per esempio, provare a fare quanto segue.

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

Eseguire qualsiasi comando di shell sotto `script`.

Premere `Ctrl-D` per uscire da `script`.

```
$ vim typescript
```

Vedere Sezione [9.1.1](#).

1.4.10 Comandi Unix di base

È bene imparare i comandi Unix di base. Il termine "Unix" è qui usato in senso lato; ogni SO clone di Unix offre di solito comandi equivalenti. Il sistema Debian non fa eccezione. Non ci si preoccupi se, al momento, alcuni comandi non funzionano come si vorrebbe. Se si usa `alias` nella shell, i corrispondenti output dei comandi sono diversi. Questi esempi non sono pensati per essere eseguiti necessariamente in questo ordine.

Provare tutti i comandi seguenti da un account utente non privilegiato.

Nota

Unix tradizionalmente nasconde i nomi di file che iniziano con ".". Sono tradizionalmente file contenenti informazioni di configurazione e preferenze dell'utente.

Per il comando `cd` vedere `builtins(7)`.

Il paginatore predefinito del sistema di base di Debian è `more(1)` che non ha lo scorrimento all'indietro. Installando il pacchetto `less`, con la riga di comando `apt-get install less`, `less(1)` diventa il paginatore predefinito e si può così scorrere il testo all'indietro usando i tasti freccia.

I caratteri "[" e "]" nella espressione regolare del comando `ps aux | grep -e "[e]xim4*"` sopra citato, permettono a `grep` di evitare di trovare corrispondenza con sé stesso. La parte "4*" nella espressione regolare significa 0 o più ripetizioni del carattere "4" e perciò permette a `grep` di trovare corrispondenza sia con "exim" sia con "exim4". Sebbene il carattere "*" sia usato nei nomi di file con metacaratteri della shell e nelle espressioni regolari, il suo significato è diverso nei due casi. Si possono imparare le espressioni regolari da `grep(1)`.

Come esercizio, esplorare le directory e dare un'occhiata al sistema usando i comandi citati sopra. Se si hanno domande su uno qualsiasi dei comandi in console, assicurarsi di leggere la pagina di manuale.

Per esempio, provare a fare quanto segue.

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

Può essere un po' difficile abituarsi allo stile delle pagine man perché sono piuttosto concise, in particolar modo le più vecchie e tradizionali. Una volta che ci si fa la mano, però, si apprezza la loro concisione.

Notare che molti comandi in stile Unix, inclusi quelli da GNU e BSD, mostrano una breve informazione di aiuto se li si invoca in uno dei modi seguenti (o in alcuni casi se lo si fa senza usare opzioni).

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

1.5 Il semplice comando di shell

Ora si ha un'idea di come usare il sistema Debian; è tempo di dare uno sguardo più approfondito al meccanismo di esecuzione dei comandi nel sistema Debian. Qui la realtà è stata semplificata ad uso del principiante. Vedere `bash(1)` per una spiegazione esatta.

Un semplice comando è formato da una sequenza di componenti.

comando	descrizione
<code>pwd</code>	mostra il nome della directory attuale/di lavoro
<code>whoami</code>	mostra il nome dell'utente attuale
<code>id</code>	mostra l'identità dell'utente attuale (nome, uid, gid e gruppi associati)
<code>file pippo</code>	mostra che tipo di file sia il file "pippo"
<code>type -p nomecomando</code>	mostra la posizione del file del comando "nomecomando"
<code>which nomecomando</code>	" "
<code>type nomecomando</code>	mostra informazioni sul comando "nomecomando"
<code>apropos parola-chiave</code>	trova comandi riguardanti "parola-chiave"
<code>man -k parola-chiave</code>	" "
<code>whatis nomecomando</code>	mostra una spiegazione di una riga sul comando "nomecomando"
<code>man -a nomecomando</code>	mostra una spiegazione del comando "nomecomando" (in stile Unix)
<code>info nomecomando</code>	mostra una spiegazione piuttosto lunga del comando "nomecomando" (in stile GNU)
<code>ls</code>	elenca il contenuto di directory (non i file punto e le directory)
<code>ls -a</code>	elenca il contenuto di directory (tutti i file e directory)
<code>ls -A</code>	elenca il contenuto di directory (quasi tutti i file e directory, cioè salta ". ." e ".")
<code>ls -la</code>	elenca tutto il contenuto di directory con informazioni dettagliate
<code>ls -lai</code>	elenca tutto il contenuto di directory con numeri di inode e informazioni dettagliate
<code>ls -d</code>	elenca tutte le directory dentro la directory attuale
<code>tree</code>	mostra il contenuto in forma di albero
<code>lsuf pippo</code>	mostra lo stato aperto per il file "pippo"
<code>lsuf -p pid</code>	mostra i file aperti dal processo con ID "pid"
<code>mkdir pippo</code>	crea una nuova directory "pippo" nella directory attuale
<code>rmdir pippo</code>	rimuove la directory "pippo" nella directory attuale
<code>cd pippo</code>	cambia directory spostandosi nella directory "pippo" nella directory attuale o in una directory elencata nella variabile "\$CDPATH"
<code>cd /</code>	cambia directory spostandosi nella directory radice
<code>cd</code>	cambia directory spostandosi nella directory home dell'utente
<code>cd /pippo</code>	cambia directory spostandosi nella directory con percorso assoluto "/pippo"
<code>cd ..</code>	cambia directory spostandosi nella directory genitore
<code>cd ~pippo</code>	cambia directory spostandosi nella directory home dell'utente "pippo"
<code>cd -</code>	cambia directory spostandosi nella directory precedente
<code></etc/motd pager</code>	mostra il contenuto di "/etc/motd" usando il paginatore predefinito
<code>touch filediprova</code>	crea un file "filediprova" vuoto
<code>cp pippo pluto</code>	copia un file "pippo" esistente in un nuovo file "pluto"
<code>rm filediprova</code>	rimuove il file "filediprova"
<code>mv pippo pluto</code>	rinomina un file "pippo" esistente con il nuovo nome "pluto" ("pluto" non deve esistere)
<code>mv pippo pluto</code>	muove un file "pippo" esistente nella nuova posizione "pluto/pippo" (la directory "pluto" deve esistere)
<code>mv pippo pluto/paperino</code>	muove un file "pippo" esistente in una nuova posizione con il nuovo nome "pluto/paperino" (la directory "pluto" deve esistere ma non deve esistere "bar/baz")
<code>chmod 600 pippo</code>	fa sì che il file "pippo" esistente sia non leggibile e non scrivibile da altri (e non eseguibile per nessuno)
<code>chmod 644 pippo</code>	fa sì che il file "pippo" esistente sia leggibile ma non scrivibile da altri (e non eseguibile per nessuno)
<code>chmod 755 pippo</code>	fa sì che il file "pippo" esistente sia leggibile ma non scrivibile da altri (ed eseguibile per tutti)
<code>find . -name modello</code>	trova nomi di file corrispondenti al "modello" di shell (lento)
<code>locate -d . modello</code>	trova nomi di file corrispondenti al "pattern" di shell (più veloce, usa un database generato regolarmente)
<code>grep -e "modello" *.html</code>	trova un "modello" in tutti i file nella directory attuale che terminano con ".html" e mostra tutte le corrispondenze
<code>top</code>	mostra informazioni sui processi a tutto schermo, digitare "q" per uscire
<code>ps aux pager</code>	mostra informazioni su tutti i processi in esecuzione usando output in stile BSD
<code>ps -ef</code>	mostra informazioni su tutti i processi in esecuzione usando output in

1. Assegnazioni di variabili (opzionale)
2. Nome del comando
3. Opzioni (opzionali)
4. Ridirezioni (opzionali, > , >> , < , << , ecc.)
5. Operatori di controllo (opzionale, && , | | , *a-capo* , ; , & , (,))

1.5.1 Esecuzione dei comandi e variabili d'ambiente

I valori di alcune [variabili d'ambiente](#) cambiano il comportamento di alcuni comandi Unix.

I valori predefiniti delle variabili d'ambiente sono impostati inizialmente dal sistema PAM e poi alcuni di essi possono essere reimpostati da alcuni programmi applicativi.

- 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 La variabile "\$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.

Il valore completo di localizzazione impostato nella variabile "\$LANG" consiste di 3 parti: "xx_YY.ZZZZ".

valore di localizzazione	significato
xx	codice ISO 639 della lingua (in minuscole), come "en"
YY	codice ISO 3166 del paese (in maiuscole) come "US"
ZZZZ	codeset, impostato sempre ad "UTF-8"

Tabella 1.18: Le 3 parti del valore di localizzazione

Una tipica esecuzione di un comando usa una sequenza sulla riga della shell come la seguente.

```
$ 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.

- Per il primo comando "\$LANG" è impostata al valore di [localizzazione](#) predefinito di sistema: "en_US.UTF-8".
-

localizzazione raccomandata	lingua (zona)
en_US.UTF-8	inglese (USA)
en_GB.UTF-8	inglese (Gran Bretagna)
fr_FR.UTF-8	francese (Francia)
de_DE.UTF-8	tedesco (Germania)
it_IT.UTF-8	italiano (Italia)
es_ES.UTF-8	spagnolo (Spagna)
ca_ES.UTF-8	catalano (spagna)
sv_SE.UTF-8	svedese (Svezia)
pt_BR.UTF-8	portoghese (Brasile)
ru_RU.UTF-8	russo (Russia)
zh_CN.UTF-8	cinese (Repubblica Popolare Cinese)
zh_TW.UTF-8	cinese (Taiwan Repubblica di Cina)
ja_JP.UTF-8	giapponese (Giappone)
ko_KR.UTF-8	coreano (Corea del Sud)
vi_VN.UTF-8	vietnamita (Vietnam)

Tabella 1.19: Elenco di localizzazioni raccomandate

- Per il secondo comando "\$LANG" è impostata al valore di [localizzazione](#) UTF-8 francese: `fr_FR.UTF-8`.

La maggior parte delle invocazioni di comandi non è solitamente preceduta da definizioni di variabili d'ambiente. In alternativa all'esempio precedente, si può eseguire quanto segue.

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

Suggerimento

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.

Per dettagli più specifici sulla configurazione della localizzazione vedere Sezione [8.1](#).

1.5.3 La variabile "\$PATH"

Quando si digita un comando nella shell, questa cerca il comando nelle directory nell'elenco contenuto nella variabile d'ambiente "\$PATH". Il valore della variabile d'ambiente "\$PATH" è anche chiamato percorso di ricerca della shell.

In una installazione Debian standard, la variabile d'ambiente "\$PATH" degli account utente può non includere `/sbin` e `/usr/sbin`. Il comando `ifconfig`, per esempio, deve essere eseguito con il percorso completo, come in `/sbin/ifconfig`. (Il comando simile `ip` si trova in `/bin`.)

Si può modificare la variabile d'ambiente "\$PATH" della shell Bash tramite il file `~/ .bash_profile` o `~/ .bashrc`.

1.5.4 La variabile "\$HOME"

Molti comandi memorizzano una configurazione specifica per l'utente nella directory home e cambiano il loro comportamento in base al suo contenuto. La directory home è identificata dalla variabile d'ambiente "\$HOME".

Suggerimento

La shell espande `~/` nella directory home dell'utente attuale, cioè `$HOME/`. La shell espande `~/pippo/` nella directory home di pippo, cioè `/home/pippo/`.

valore di "\$HOME"	situazione di esecuzione del programma
/	programma eseguito da processo init (demone)
/root	programma eseguito dalla normale shell di root
/home/utente_normale	programma eseguito dalla shell di un utente normale
/home/utente_normale	programma eseguito dal menu del desktop grafico di un utente normale
/home/utente_normale	programma eseguito come root con "sudo programma"
/root	programma eseguito come root con "sudo -H programma"

Tabella 1.20: Elenco di valori di "\$HOME"

1.5.5 Opzioni della riga di comando

Alcuni comandi accettano argomenti. Gli argomenti che iniziano con "-" o "--" sono chiamati opzioni e controllano il comportamento del comando.

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

In questo caso l'opzione "-R" sulla riga di comando cambia il comportamento di `date(1)` producendo in output la data in una stringa conforme alla [RFC2822](https://tools.ietf.org/html/rfc2822).

1.5.6 Glob della shell

Spesso si desidera che un comando abbia effetto su un gruppo di file senza dover digitarli tutti. I modelli di espansione dei nomi di file che usano i **glob** della shell, a volte detti **metacaratteri** o **caratteri jolly**, facilitano questo compito.

modello di glob della shell	descrizione della regola di corrispondenza
*	nome file (segmento) che non inizia con "."
.*	nome file (segmento) che inizia con "."
?	esattamente un carattere
[...]	esattamente un carattere tra quelli racchiusi tra le parentesi quadre
[a-z]	esattamente un carattere che sia un carattere compreso tra "a" e "z"
[^...]	esattamente un carattere che non sia uno di quelli racchiusi tra parentesi quadre (non contando "^")

Tabella 1.21: Modelli di glob della shell

Per esempio, provare a fare quanto segue.

```
$ 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
```

Vedere `glob(7)`.

Nota

A differenza della normale espansione dei nomi della shell, il modello shell "*" cercato dal comando `find(1)` con la ricerca "- name" ecc., trova corrispondenza anche con nomi di file che iniziano con ".". (Nuova funzionalità [POSIX](#).)

Nota

Si può manipolare il comportamento di BASH in relazione ai modelli di glob usando le sue opzioni shopt incorporate come `dotglob`, `noglob`, `nocaseglob`, `nullglob`, `extglob`, ecc. Vedere `bash(1)`.

1.5.7 Valore restituito dal comando

Ogni comando restituisce il suo stato di uscita (variabile "\$?") come valore restituito.

stato di uscita del comando	valore numerico restituito	valore logico restituito
successo	zero, 0	VERO
errore	non-zero, -1	FALSO

Tabella 1.22: Codici di uscita dei comandi

Per esempio provare quanto segue.

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

Nota

Notare che, nel contesto logico della shell, il **successo** è trattato come il valore logico **VERO** che ha valore numerico 0 (zero). Questo è in qualche modo non intuitivo ed è bene rimarcarlo.

1.5.8 Sequenze tipiche di comandi e ridirezione della shell

È bene cercare di ricordare i seguenti idiomi per comandi di shell che vengono digitati sulla stessa riga come parte del comando di shell.

Il sistema Debian è un sistema multi-tasking. I compiti sullo sfondo permettono agli utenti di eseguire più programmi in una singola shell. La gestione dei processi sullo sfondo coinvolge i comandi interni della shell `jobs`, `fg`, `bg` e `kill`. Leggere le sezioni "SIGNALS" e "JOB CONTROL" di `bash(1)` e `builtin(1)`.

Per esempio, provare a fare quanto segue.

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

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

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

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

idioma per comando	descrizione
comando &	esegue comando sullo sfondo in una sotto-shell
comando1 comando2	invia tramite pipe lo standard output di comando1 allo standard input di comando2 (esecuzione concorrente)
comando1 2>&1 comando2	invia tramite pipe sia lo standard output sia lo standard error di comando1 allo standard input di comando2 (esecuzione concorrente)
comando1 ; comando2	esegue comando1 e comando2 in sequenza
comando1 && comando2	esegue comando1; se ha successo, esegue comando2 in sequenza (restituisce lo stato di successo se sia comando1 sia comando2 hanno successo)
comando1 comando2	esegue comando1; se non ha successo, esegue comando2 in sequenza (restituisce lo stato di successo se comando1 o comando2 ha successo)
comando > pippo	ridirige lo standard output di comando al file "pippo" (sovrascrivendolo)
comando 2> pippo	ridirige lo standard error di comando al file "pippo" (sovrascrivendolo)
comando >> pippo	ridirige lo standard output di comando al file "pippo" (aggiungendo in coda)
comando 2>> pippo	ridirige lo standard error di comando al file "pippo" (aggiungendo in coda)
comando > pippo 2>&1	ridirige sia lo standard output sia lo standard error di comando al file pippo
comando < pippo	usa come standard input di comando il file "pippo"
comando << delimitatore	usa come standard input di comando le righe che seguono fino a che non viene incontrato il testo "delimitatore" (inserire un documento)
comando <<- delimitatore	usa come standard input di comando le righe che seguono fino a che non viene incontrato il testo "delimitatore"; i caratteri di tabulazione iniziali vengono eliminate dal testo (inserire un documento)

Tabella 1.23: Idiomi per comandi di shell

Benché tutti i 4 esempi di ridirezione della shell mostrino la stessa cosa, l'ultimo esempio esegue un comando `cat` aggiuntivo e spreca risorse senza motivo.

La shell permette di aprire file usando il comando interno `exec` con un descrittore di file arbitrario.

```
$ 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
```

I descrittori di file 0-2 sono predefiniti.

device	descrizione	descrittore del file
<code>stdin</code>	standard input	0
<code>stdout</code>	standard output	1
<code>stderr</code>	standard error	2

Tabella 1.24: Descrittori di file predefiniti

1.5.9 Alias di comandi

Si possono impostare alias per i comandi usati di frequente.

Per esempio, provare a fare quanto segue.

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

Ora `la` funziona da scorciatoia per `ls -la` che elenca tutti i file nel formato di elencazione esteso.

Si possono elencare tutti gli alias esistenti con `alias` (vedere la sezione "SHELL BUILTIN COMMANDS" in `bash(1)`).

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

Si può identificare il percorso esatto o l'identità di un comando con `type` (vedere la sezione "SHELL BUILTIN COMMANDS" di `bash(1)`).

Per esempio, provare a fare quanto segue.

```
$ 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
```

In questo caso `ls` era stato cercato di recente mentre `file` no, perciò `ls` risulta "hashed", cioè la shell ha un record interno per un accesso veloce alla posizione del comando `ls`.

Suggerimento

Vedere Sezione [9.3.6](#).

1.6 Elaborazione di testo stile Unix

Nell'ambiente di lavoro Unix, l'elaborazione del testo viene fatta instradando il testo tramite pipe attraverso catene di strumenti standard per l'elaborazione del testo. Questa è stata un'altra fondamentale innovazione di Unix.

1.6.1 Strumenti Unix per il testo

Ci sono alcuni strumenti di elaborazione del testo che sono usati molto spesso in un sistema *nix.

- Senza l'uso di alcuna espressione regolare:
 - `cat(1)` concatena file e produce in output il contenuto totale.
 - `tac(1)` concatena file e produce l'output in senso inverso.
 - `cut(1)` seleziona parti di righe e le produce in output.
 - `head(1)` produce in output la prima parte di file.
 - `tail(1)` produce in output l'ultima parte di file.
 - `sort(1)` ordina righe in file di testo.
 - `uniq(1)` rimuove righe duplicate da un file ordinato.
 - `tr(1)` trasforma o elimina caratteri.
 - `diff(1)` paragona file riga per riga.
- Basic regular expression (**BRE**) is used as default:
 - `ed(1)` è un primitivo editor di riga.
 - `sed(1)` è un editor di flussi.
 - `grep(1)` trova corrispondenze di testo con modelli.
 - `vim(1)` è un editor a schermo.
 - `emacs(1)` è un editor a schermo (con **BRE** un po' estese)
- Con l'uso di espressioni regolari estese (**ERE**):
 - `awk(1)` fa semplici elaborazioni di testo.
 - `egrep(1)` trova corrispondenze di testo con modelli.
 - `tc(3tbl)` può fare ogni elaborazione di testo immaginabile: vedere `re_syntax(3)`. Spesso usato con `tk(3tk)`.
 - `perl(1)` può fare ogni elaborazione di testo immaginabile. Vedere `perlre(1)`.
 - `pcgrep(1)` dal pacchetto `pcgrep` trova corrispondenze di testo con modelli [PCRE \(Perl Compatible Regular Expressions\)](#).
 - `python(1)` con il modulo `re` può fare ogni elaborazione di testo immaginabile. Vedere `"/usr/share/doc/python/html/in`

Se non si è sicuri di cosa facciano esattamente questi comandi, usare `"man comando"` per scoprirlo.

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 Sezione [8.1](#)).

Nota

Le espressioni regolari [Perl](#) (`perlre(1)`), le [PCRE \(Perl Compatible Regular Expressions\)](#) e le espressioni regolari [Python](#) fornite dal modulo `re` hanno molte estensioni comuni delle normali **ERE**.

1.6.2 Espressioni regolari

Le **espressioni regolari** sono usate in molti strumenti di elaborazione del testo. Sono analoghe ai modelli di glob della shell, ma sono più complesse e potenti.

Una espressione regolare descrive il modello a cui trovare corrispondenza ed è composta da caratteri di testo e **metacaratteri**.

Un **metacarattere** è semplicemente un carattere con un significato speciale. Ci sono 2 stili principali, **BRE** e **ERE**, a seconda degli strumenti di testo descritti in precedenza.

BRE	ERE	descrizione della espressione regolare
\ . [] ^ \$ *	\ . [] ^ \$ *	metacaratteri comuni
\+ \? \(\) \{ \} \		metacaratteri protetti dal carattere "\" di escape solo di BRE
	+ ? () { }	metacaratteri non protetti dal carattere "\" di escape solo di ERE
c	c	corrisponde al non metacarattere "c"
\c	\c	corrisponde al carattere letterale "c" anche se "c" è in sé stesso un metacarattere
.	.	corrisponde a qualsiasi carattere incluso il carattere di a capo
^	^	posizione all'inizio di una stringa
\$	\$	posizione alla fine di una stringa
\<	\<	posizione all'inizio di una parola
\>	\>	posizione alla fine di una parola
[abc...]	[abc...]	corrisponde ad ogni carattere in "abc..."
[^abc...]	[^abc...]	corrisponde ad ogni carattere eccetto a quelli in "abc..."
r*	r*	corrisponde a zero o più occorrenze della espressione regolare identificata da "r"
r\+	r+	corrisponde ad una o più occorrenza della espressione regolare identificata da "r"
r\?	r?	corrisponde a zero o una occorrenza dell'espressione regolare identificata da "r"
r1 r2	r1 r2	corrisponde ad una occorrenza della espressione regolare identificata da "r1" o "r2"
\(r1 r2\)	(r1 r2)	corrisponde ad una occorrenza dell'espressione regolare identificata da "r1" o "r2" e la tratta come una espressione regolare tra parentesi

Tabella 1.25: Metacaratteri per BRE e ERE

Le espressioni regolari di **emacs** sono fondamentalmente **BRE**, ma sono state estese per trattare "+" e "?" come i **metacaratteri** nelle **ERE**. Perciò, nelle espressioni regolari di **emacs**, non è necessario proteggerli con il carattere di escape "\".

Per effettuare ricerche di testo usando espressioni regolari si può usare **grep(1)**.

Per esempio, provare a fare quanto segue.

```
$ 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
```

Suggerimento

Vedere Sezione [9.3.6](#).

1.6.3 Espressioni di sostituzione

Per le espressioni di sostituzione alcuni caratteri hanno un significato particolare.

espressione di sostituzione	descrizione del testo che sostituirà l'espressione di sostituzione
&	ciò che ha avuto corrispondenza con l'espressione regolare (usare <code>\&</code> in <code>emacs</code>)
<code>\n</code>	ciò che ha avuto corrispondenza con la <code>n</code> -esima espressione regolare tra parentesi (" <code>n</code> " è un numero)

Tabella 1.26: L'espressione di sostituzione

Per le stringhe di sostituzione Perl viene usato «`$&`» invece di «`&`» e «`$n`» invece di «`\n`».

Per esempio, provare a fare quanto segue.

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

In questo esempio si faccia particolare attenzione allo stile delle espressioni regolari **tra parentesi** e come le stringhe corrispondenti siano usate nel processo di sostituzione del testo dai diversi strumenti.

Queste espressioni regolari possono essere usate per spostare il cursore e per fare sostituzioni di testo anche in alcuni editor.

La barra inversa, "`\`" alla fine della prima riga nella riga di comando di shell protegge il carattere di a capo rendendolo uno spazio e continuando l'input della riga di comando di shell nella riga successiva.

Leggere tutte le pagine man relative per imparare l'uso di questi comandi.

1.6.4 Sostituzione globale con espressioni regolari

Il comando `ed(1)` può sostituire tutte le occorrenze di "`DA_REGEX`" con "`A_TESTO`" in "`file`".

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

Il comando `sed(1)` può sostituire tutte le occorrenze di "`DA_REGEX`" con "`A_TESTO`" in "`file`".

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

Il comando `vim(1)` può sostituire tutte le occorrenze di "`DA_REGEX`" with "`A_TESTO`" in "`file`" usando il comando `ex(1)`.

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

Suggerimento

Il modificatore "`c`" nell'esempio soprastante assicura che venga chiesta la conferma interattiva per ciascuna sostituzione.

Si possono elaborare file multipli ("file1", "file2" e "file3") in modo simile con vim(1) o perl(1).

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

Suggerimento

Il modificatore "e" nell'esempio soprastante evita che un errore di nessuna corrispondenza trovata ("No match") rompa il processo.

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

Nell'esempio perl(1), "-i" è per la modifica in-situ di ciascun file interessato, e "-p" è per reiterare implicitamente il comando su tutti i file specificati.

Suggerimento

L'uso dell'opzione "-i.bak" invece di "-i" preserva una copia del file originale aggiungendo il suffisso ".bak" al suo nome file. Questo permette di ripristinare le cose più facilmente in caso di errori in sostituzioni complesse.

Nota

ed(1) e vim(1) usano BRE; perl(1) usa ERE.

1.6.5 Estrarre dati da file con tabelle di testo

Considerare per esempio un file di testo chiamato "DPL" in cui sono elencati, in un formato separato da spazi, alcuni nomi di leader del progetto Debian pre-2004 e la data della loro installazione.

```
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
```

Suggerimento

Vedere ["Una breve storia di Debian"](#) per la [storia della Guida del progetto Debian](#) aggiornata.

Awk viene usato spesso per estrarre dati da questo tipo di file.

Per esempio, provare a fare quanto segue.

```
$ 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
```

Questo tipo di file può anche essere analizzato con le shell tipo Bash.

Per esempio, provare a fare quanto segue.

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

In questo caso, il comando incorporato `read` usa i caratteri in `"$IFS"` (internal field separator, separatore di campi interno) per suddividere le righe in parole.

Se si modifica `"$IFS"` in `":,"`, si possono analizzare facilmente con la shell i file `"/etc/passwd"`.

```
$ 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 si usa `Awk` per fare la stessa cosa, usare `"FS=':,'"` per impostare il separatore di campi.)

`IFS` viene anche usato dalla shell per suddividere i risultati dell'espansione di parametri, sostituzione di comandi ed espansioni aritmetiche. Queste non avvengono all'interno delle virgolette singole o doppie. Il valore predefinito di `IFS` è la combinazione di *spazio*, *tab* e *<a capo>*.

Essere prudenti nell'uso di questi trucchetti di shell con `IFS`. Possono accadere strane cose quando la shell interpreta alcune parti dello script come suo **input**.

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

1.6.6 Frammenti di script per comandi con pipe

Gli script seguenti fanno alcune cose carine se inseriti in una pipe.

Uno script di shell di una riga può operare reiteratamente su più file usando `find(1)` e `xargs(1)` per fare compiti piuttosto complicati. Vedere Sezione [10.1.5](#) e Sezione [9.4.9](#).

Quando usare la shell in modalità interattiva diventa troppo complicato, considerare la scrittura di uno script di shell (vedere Sezione [12.1](#)).

frammento di script (digitarlo in un'unica riga)	effetto del comando
<code>find /usr -print</code>	trova tutti i file in <code>"/usr"</code>
<code>seq 1 100</code>	stampa da 1 a 100
<code> xargs -n 1 comando</code>	esegue ripetutamente il comando con ogni elemento dalla pipe come argomento
<code> xargs -n 1 echo</code>	suddivide elementi separati da spazio nella pipe su righe distinte
<code> xargs echo</code>	unisce tutte le righe dalla pipe in una riga
<code> grep -e modello_regex</code>	estrae dalla pipe le righe che contengono <code>modello_regex</code>
<code> grep -v -e modello_regex</code>	estrae dalla pipe le righe che non contengono <code>modello_regex</code>
<code> cut -d: -f3 -</code>	estrae dalla il terzo campo, separato da ":" (file passwd, ecc.)
<code> awk '{ print \$3 }'</code>	estrae dalla pipe il terzo campo separato da spazi
<code> awk -F\t '{ print \$3 }'</code>	estrae dalla pipe il terzo campo separato da tabulazioni
<code> col -bx</code>	rimuove i caratteri backspace ed espande le tabulazioni in sequenze di spazi
<code> expand -</code>	espande le tabulazioni
<code> sort uniq</code>	ordina e rimuove i duplicati
<code> tr 'A-Z' 'a-z'</code>	converte le maiuscole in minuscole
<code> tr -d '\n'</code>	concatena le righe in un'unica riga
<code> tr -d '\r'</code>	rimuove i caratteri di a capo
<code> sed 's/^/# /'</code>	aggiunge <code>"#"</code> all'inizio di ogni riga
<code> sed 's/\.ext//g'</code>	rimuove <code>".ext"</code>
<code> sed -n -e 2p</code>	stampa la seconda riga
<code> head -n 2 -</code>	stampa le prime due righe
<code> tail -n 2 -</code>	stampa le ultime due righe

Tabella 1.27: Elenco di frammenti di script per comandi con pipe

Capitolo 2

Gestione dei pacchetti in Debian

Nota

Questo capitolo è stato scritto assumendo che il rilascio stabile più recente abbia nome in codice: bullseye.

[Debian](#) è un'organizzazione di volontari che crea distribuzioni **coerenti** di pacchetti binari precompilati di software libero e li distribuisce dal suo archivio.

L'[archivio Debian](#) viene fornito da [molti siti mirror remoti](#) per l'accesso con i metodi HTTP e FTP. È anche disponibile come [CD-ROM/DVD](#).

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

Il sistema di gestione dei pacchetti di Debian, **se usato in modo corretto**, permette all'utente di installare dall'archivio **insiemi coerenti di pacchetti binari** sul sistema. Attualmente ci sono 63210 pacchetti disponibili per l'architettura amd64.

Il sistema di gestione dei pacchetti di Debian ha una storia ricca e fornisce molti programmi con interfacce per l'utente finale e molti metodi di backend per l'accesso agli archivi tra cui scegliere. Attualmente le scelte raccomandate sono le seguenti.

- `apt(8)` for all interactive command line operations, including package installation, removal and dist-upgrades.
- `apt-get(8)` per chiamare il sistema di gestione dei pacchetti Debian da script. È anche un'opzione di ripiego quando non è disponibile `apt` (spesso in sistemi Debian più vecchi).
- `aptitude(8)` per un'interfaccia testuale interattiva per gestire i pacchetti installati e per cercare i pacchetti disponibili.

2.1 Prerequisiti per la gestione dei pacchetti Debian

2.1.1 Configurazione dei pacchetti

Questi sono alcuni punti fondamentali per la configurazione dei pacchetti in un sistema Debian.

- La configurazione manuale fatta dall'amministratore di sistema viene rispettata. In altre parole, il sistema di configurazione dei pacchetti non fa configurazioni invadenti per ragioni di comodità.
 - Ogni pacchetto viene fornito con un proprio script di configurazione per l'interfaccia utente standard `debconf(7)`, per aiutare nel processo di configurazione iniziale del pacchetto.
 - I Debian Developer cercano di fare del loro meglio per rendere l'aggiornamento un'esperienza senza problemi grazie agli script di configurazione dei pacchetti.
-

pacchetto	popcon	dimensione	descrizione
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	pacchetto di estensione per APT per abilitare l'installazione automatica degli aggiornamenti di sicurezza
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	programmi di utilità APT: apt-extracttemplates (1), apt-ftpparchive (1) e apt-sortpkgs (1)
apt-listchanges	V:356, I:849	423	strumento di notifica dello storico dei cambiamenti dei pacchetti
apt-listbugs	V:8, I:12	465	elenca i bug critici prima di ogni installazione di APT
apt-file	V:17, I:79	90	utilità APT per ricerca di pacchetti, interfaccia a riga di comando
apt-rdepends	V:0, I:6	40	elenca le dipendenze dei pacchetti in modo ricorsivo

Tabella 2.1: Elenco degli strumenti Debian di gestione dei pacchetti

- Sono disponibili per l'amministratore di sistema le complete funzionalità di un pacchetto software, ma quelle con rischi per la sicurezza sono disabilitate nella installazione predefinita.
- Se si attiva manualmente un servizio con rischi per la sicurezza, si è responsabili del contenimento del rischio.
- L'amministratore di sistema può abilitare manualmente configurazioni esotiche; questo può creare interferenze con popolari programmi generici di supporto per la configurazione del sistema.

2.1.2 Precauzioni base



avvertimento

Non installare pacchetti da una miscela casuale di suite. Probabilmente si disgrega la coerenza tra i pacchetti che richiede una conoscenza approfondita dalla gestione del sistema, come dell'[ABI](#) del compilatore, versione delle [librerie](#), funzionalità dell'interprete, ecc.

L'amministratore di sistema [novizio](#) dovrebbe rimanere con il rilascio **stable** di Debian ed applicare solo gli aggiornamenti di sicurezza. Ciò significa che alcune delle azioni spiegate in seguito, seppur valide, sono, come precauzione, da evitare fino a che non si capisce il sistema Debian molto bene. Ecco alcune cose da ricordare.

- Non includere **testing** o **unstable** in `"/etc/apt/sources.list"`.
- Non mescolare gli archivi standard Debian con altri archivi non Debian, come quelli di Ubuntu, in `"/etc/apt/sources.list"`.
- Non creare il file `"/etc/apt/preferences"`.
- Non cambiare il comportamento degli strumenti di gestione dei pacchetti attraverso i loro file di configurazione senza capire a pieno il loro effetto.
- Non installare pacchetti presi qua e là con `"dpkg -i pacchetto_a_caso"`.
- Non installare mai pacchetti presi qua e là con `"dpkg --force-all -i pacchetto_a_caso"`.
- Non cancellare o alterare i file in `"/var/lib/dpkg/"`.
- Non sovrascrivere file di sistema, installando direttamente programmi software compilati dai sorgenti.

- Se necessario, installarli in `"/usr/local"` o `"/opt"`.

Gli effetti causati dalle azioni descritte sopra al sistema di gestione dei pacchetti di Debian portano a incompatibilità che possono lasciare il sistema in uno stato inutilizzabile.

Gli amministratori di sistema Debian seri, che gestiscono server di importanza critica, dovrebbero usare particolari precauzioni.

- Non installare alcun pacchetto, compresi quelli di aggiornamenti di sicurezza di Debian, senza averli attentamente testati con la propria particolare configurazione in condizioni sicure.
 - Alla fin fine è l'amministratore di sistema ad essere responsabile del proprio sistema.
 - La lunga tradizione di stabilità dei sistemi Debian non è di per sé una garanzia.

2.1.3 Una vita di aggiornamenti senza fine



Attenzione

Per i **server di produzione**, è raccomandata la suite `stable` con gli aggiornamenti di sicurezza. Lo stesso vale per i PC desktop su cui si investono sforzi di amministrazione limitati.

Despite my warnings above, I know many readers of this document may wish to run the newer `testing` or `unstable` suites. L'**illuminazione** data dalla lettura di ciò che segue salva l'utente dall'eterna lotta **karmica** con l'**inferno** degli aggiornamenti e gli fa raggiungere il **nirvana** 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.

- Impostare il sistema per avere **due partizioni avviabili** installando la suite `stable` di Debian in un'altra partizione
- Tenere a portata di mano il CD di installazione per l'**avvio di ripristino**
- Considerare l'installazione di `apt-listbugs` per controllare le informazioni del **Sistema Debian di tracciamento dei Bug (BTS)** prima degli aggiornamenti
- Imparare l'infrastruttura del sistema dei pacchetti abbastanza bene da poter aggirare il problema.
- Install a corresponding sandboxed upstream binary package in case of trouble (see Sezione 7.6)
- Creare un ambiente chroot o simile ed eseguire prima lì il sistema più recente (vedere Sezione 9.11)



Attenzione

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 Nozioni di base sugli archivi Debian

Guardiamo l'[archivio Debian](#) dalla prospettiva dell'utente.

Suggerimento

La politica ufficiale per gli archivi Debian è definita nel [Capitolo 2 - "L'archivio Debian"](#), del manuale [Debian Policy](#).

Per il normale accesso HTTP, l'archivio viene specificato nel file `"/etc/apt/sources.list"` come nell'esempio seguente per l'attuale sistema `stable = bullseye`.

```
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
```

Viene qui usato il nome in codice `"bullseye"` invece del nome della suite `"stable"` per evitare sorprese una volta che venga rilasciata la nuova `stable`.

La struttura di `"/etc/apt/sources.list"` è descritta in `sources.list(5)` ed i punti chiave sono i seguenti.

- Le righe `"deb"` definiscono voci per pacchetti binari.
- Le righe `"deb-src"` definiscono voci per i pacchetti sorgenti.
- Il primo argomento è l'URL radice dell'archivio Debian.
- Il secondo argomento è il nome della distribuzione, o il nome della suite o il nome in codice.
- Il terzo argomento e quelli seguenti sono l'elenco dei nomi di aree valide dell'archivio Debian.

Le righe `"deb-src"` possono essere tranquillamente omesse (o commentate aggiungendo il carattere `"#"` all'inizio della riga) se si usa solo `aptitude` che non accede ai metadati relativi ai sorgenti. Si velocizza così l'aggiornamento dei metadati dell'archivio. L'URL può essere `"http://"`, `"ftp://"`, `"file://"`, ...

Suggerimento

Se nell'esempio precedente si usa `"sid"` invece di `"bullseye"`, in `"/etc/apt/sources.list"` non è richiesta la riga `"deb: http://security.debian.org/ ..."` per gli aggiornamenti di sicurezza. Ciò è dovuto al fatto che non esiste un archivio per gli aggiornamenti di sicurezza per `"sid"` (`unstable`).

Ecco l'elenco dei siti degli archivi Debian e dei nomi di suite e nomi in codice usati nel file di configurazione.

Attenzione



Solo il rilascio **stable** puro con gli aggiornamenti di sicurezza fornisce la massima stabilità. Avere un sistema basato per lo più sul rilascio **stable** con mescolati alcuni pacchetti dai rilasci **testing** o **unstable** è più rischioso che avere un sistema basato su un puro rilascio **unstable** per ciò che riguarda conflitti tra le versioni di libreria, ecc. Se si ha veramente bisogno, nel rilascio **stable**, dell'ultima versione di qualche programma usare pacchetti dai servizi [bullseye-updates](#) e <http://backports.debian.org> (vedere Sezione [2.7.4](#)). Questi servizi devono essere usati con estrema cautela.

Attenzione



Fondamentalmente si dovrebbe mettere una sola tra le suite `stable`, `testing` o `unstable` nelle righe `"deb"`. Se si elenca una combinazione delle suite `stable`, `testing` e `unstable` nelle righe `"deb"`, il programma APT verrà rallentato e solo l'archivio più recente avrà effetto. Elencare più di una voce ha senso quando viene usato il file `"/etc/apt/preferences"` con scopi ben precisi (vedere Sezione [2.7.3](#)).

URL dell'archivio	nome suite (nome in codice)	scopo
http://deb.debian.org/debian/	stable (bullseye)	rilascio stabile (bullseye)
http://deb.debian.org/debian/	testing (bookworm)	rilascio testing (bookworm)
http://deb.debian.org/debian/	unstable (sid)	rilascio unstable (sid)
http://deb.debian.org/debian/	experimental	pre-rilascio sperimentale (opzionale, solo per sviluppatori)
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)

Tabella 2.2: Elenco dei siti con l'archivio Debian

Suggerimento

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

I bug che riguardano la sicurezza per l'archivio stable vengono risolti dal Debian Security Team, la cui attività è piuttosto rigorosa ed affidabile. I bug per l'archivio testing possono essere risolti dal Debian Security Team. Per [svariate ragioni](#) questa attività non è così rigorosa come quella per stable e potrebbe essere necessario attendere la migrazione di pacchetti unstable corretti. I bug per l'archivio unstable sono corretti dal manutentore del pacchetto; i pacchetti unstable attivamente mantenuti sono solitamente in una forma piuttosto buona dato che sfruttano le più recenti correzioni a monte relative alla sicurezza. Per informazioni sul modo in cui Debian gestisce i bug relativi alla sicurezza, vedere le [FAQ Debian sulla sicurezza](#).

area	numero di pacchetti	criterio dei componenti del pacchetto
main	62089	aderenti alle DFSG e senza dipendenze da non-free
contrib	351	aderenti alle DFSG ma con dipendenze da non-free
non-free	770	non aderenti alle DFSG

Tabella 2.3: Elenco delle aree dell'archivio Debian

Nella tabella soprastante il numero dei pacchetti è per l'architettura amd64. L'area main fornisce il sistema Debian (vedere Sezione 2.1.5).

L'organizzazione dell'archivio Debian può essere studiata meglio puntando il proprio browser a ciascuno degli URL dell'archivio con aggiunto in coda dists o pool.

Ci si riferisce alla distribuzione in due modi, con la suite o con il [nome in codice](#). In alternativa la parola distribuzione viene usata come sinonimo di suite in molta documentazione. La relazione tra la suite e il nome in codice può essere riassunta nel modo seguente.

La storia dei nomi in codice è descritta nelle [FAQ Debian: 6.2.1 Quali altri nomi in codice sono stati usati in passato?](#)

Nella terminologia per gli archivi Debian più precisa, la parola "sezione" è usata specificatamente per la categorizzazione dei pacchetti in base all'area di applicazione. (Anche se l'espressione "sezione main" può essere a volte usata per descrivere l'area dell'archivio Debian chiamata "main".)

periodo	suite = stable	suite = testing	suite = unstable
dopo il rilascio bullseye	nome in codice = bullseye	nome in codice = bookworm	nome in codice = sid
dopo il rilascio bookworm	nome in codice = bookworm	nome in codice = trixie	nome in codice = sid

Tabella 2.4: Relazione tra suite e nome in codice

Ogni volta che uno sviluppatore Debian (DD) carica un nuovo pacchetto nell'archivio **unstable** (passando per [incoming](#)), gli viene richiesto di assicurare che i pacchetti caricati siano compatibili con l'insieme più recente di pacchetti nell'archivio **unstable** più recente.

Se un DD rompe questa compatibilità intenzionalmente per importanti aggiornamenti di librerie, ecc. di solito viene fatto un annuncio nella [mailing list debian-devel](#), ecc.

Prima di muovere un insieme di pacchetti dall'archivio **unstable** all'archivio **testing**, lo script di gestione degli archivi Debian non solo controlla la maturità (circa 10 giorni di età) e lo stato delle segnalazioni di bug RC per i pacchetti, ma cerca anche di assicurare che siano compatibili con il più recente insieme di pacchetti nell'archivio **testing**. Questo processo rende l'archivio **testing** molto aggiornato e usabile.

Attraverso il graduale processo di freeze dell'archivio, guidato dal team di rilascio, l'archivio **testing** viene fatto maturare con un po' di intervento manuale per renderlo completamente coerente e libero da bug. Quindi viene creato il nuovo rilascio **stable** assegnando il nome in codice per il vecchio archivio **testing** al nuovo archivio **stable** e creando il nuovo nome in codice per il nuovo archivio **testing**. Il contenuto iniziale del nuovo archivio **testing** è esattamente lo stesso dell'archivio **stable** appena rilasciato.

Sia l'archivio **unstable** sia l'archivio **testing** possono soffrire di problemi temporanei a causa di diversi fattori.

- Pacchetti difettosi caricati nell'archivio (principalmente per **unstable**)
- Ritardo nell'accettazione di nuovi pacchetti nell'archivio (principalmente per **unstable**)
- Problemi nei tempi di sincronizzazione degli archivi (sia per **testing** sia per **unstable**)
- Interventi manuali all'archivio, come rimozione di pacchetti (più per **testing**) ecc.

Perciò se si decide di usare questi archivi, si dovrebbe essere in grado di risolvere o aggirare questo tipo di problemi.

Attenzione



Per qualche mese circa dopo un nuovo rilascio **stable**, la maggior parte degli utenti desktop dovrebbe usare l'archivio **stable** con i suoi aggiornamenti di sicurezza, anche se di solito usano gli archivi **unstable** o **testing**. Durante questo periodo di transizione, entrambi gli archivi **unstable** e **testing** non sono adatti alla maggior parte degli utenti. È difficile mantenere il proprio sistema in condizioni buone di funzionamento con l'archivio **unstable** dato che è affetto da ondate di aggiornamenti importanti di pacchetti fondamentali. Anche l'archivio **testing** non è molto utile perché contiene per lo più le stesse cose dell'archivio **stable** senza il suo supporto per la sicurezza ([Debian testing-security-announce 2008-12](#)). Dopo circa un mese, l'archivio **unstable** potrebbe essere utilizzabile se si è cauti.

Suggerimento

Quando si usa l'archivio **testing**, i problemi causati da un pacchetto rimosso vengono solitamente aggirati installando il corrispondente pacchetto dall'archivio **unstable** che è caricato per risolvere un bug.

Vedere il [manuale Debian Policy](#) per le definizioni degli archivi.

- ["Sezioni"](#)

- ”Priorità”
- ”Sistema base”
- ”Pacchetti essenziali”

2.1.5 Debian è al 100% software libero

Debian è al 100% software libero perché:

- Debian installa in modo predefinito solo software libero per rispettare le libertà dell’utente.
- Debian fornisce in `main` solo software libero.
- Debian raccomanda l’esecuzione del solo software libero contenuto in `main`.
- Nessun pacchetto in `main` dipende o raccomanda pacchetti in `non-free` o `contrib`.

Alcune persone si chiedono se i seguenti due fatti siano in contraddizione o meno tra loro.

- «Debian rimarrà libera al 100%». (Prima voce del [Contratto sociale Debian](#))
- I server Debian ospitano alcuni pacchetti `non-free` e `contrib`.

Queste due cose non sono in contraddizione, perché:

- Il sistema Debian è libero al 100% e i suoi pacchetti sono ospitati dai server Debian nell’area `main`.
- I pacchetti esterni al sistema Debian sono ospitati dai server Debian nelle aree `non-free` e `contrib`.

Queste sono spiegate in dettaglio nella quarta e nella quinta voce del [Contratto sociale Debian](#):

- Le nostre priorità sono gli utenti ed il software libero
 - Ci faremo guidare dai bisogni dei nostri utenti e della comunità del software libero. Metteremo al primo posto i loro interessi. Supporteremo le necessità dei nostri utenti di operare in molti diversi tipi di ambienti di calcolo. Non ci opporremo alle opere non libere che siano state pensate per l’uso in sistemi Debian e non richiederemo compensi a chi crea o usa queste opere. Permetteremo ad altri di creare distribuzioni contenenti sia il sistema Debian che altre opere, senza richiedere compensi. Per raggiungere questi scopi, forniremo un sistema integrato di materiali di alta qualità senza alcuna restrizione legale che limiti qualsiasi uso del sistema.
- Opere che non rispettano i nostri standard free software
 - Ci rendiamo conto che alcuni dei nostri utenti richiedono di usare opere non conformi alle Debian Free Software Guidelines. Abbiamo creato le aree «contrib» e «non-free» nel nostro archivio per queste opere. I pacchetti in queste aree non fanno parte del sistema Debian, sebbene siano stati configurati per l’uso con Debian. Invitiamo i realizzatori di CD a leggere le licenze dei pacchetti in queste aree per determinare se possono distribuire i pacchetti sui loro CD. Inoltre, anche se le opere non libere non fanno parte di Debian, supporteremo il loro uso e forniremo infrastrutture per i pacchetti non liberi (come il nostro bug tracking system e le mailing list).

Gli utenti dovrebbero essere a conoscenza dei rischi correlati all’uso di pacchetti nelle aree `non-free` e `contrib`:

- mancanza di libertà legata a tali pacchetti software
 - mancanza di supporto da parte di Debian per tali pacchetti software (Debian non può supportare del software in maniera adeguata senza avere accesso al suo codice sorgente)
 - contaminazione del sistema Debian libero al 100%
-

Il documento [Debian Free Software Guidelines](#) è lo standard di [Debian](#) per il software libero. Debian interpreta la parola "software" nel suo senso più ampio includendo documentazione, firmware, logo e dati artistici nel pacchetto. Questo fa sì che gli standard Debian per il software libero siano molto severi.

I pacchetti `non-free` e `contrib` tipici includono pacchetti liberamente distribuibili dei seguenti tipi:

- Pacchetti di documentazione distribuiti sotto la [GNU Free Documentation License](#) con sezioni invariati come quelli per GCC e Make (per lo più presenti nella sezione `non-free/doc`).
- Pacchetti di firmware contenenti dati binari senza sorgenti come quelli elencati in Sezione [9.10.5](#) come non liberi (per lo più presenti nella sezione `non-free/kernel`).
- Pacchetti di giochi e tipi di carattere con restrizioni sull'uso a scopo commerciale o sulla modifica dei contenuti.

Notare che il numero dei pacchetti `non-free` e `contrib` è meno del 2% di quello dei pacchetti in `main`. Permettere l'accesso alle aree `non-free` e `contrib` non oscura la fonte dei pacchetti. L'uso interattivo a schermo intero di `aptitude(8)` fornisce piena visibilità e controllo su quali pacchetti vengano installati e da quali aree, per permettere di mantenere il proprio sistema libero quanto lo si desidera.

2.1.6 Dipendenze dei pacchetti

Il sistema Debian offre un insieme coerente di pacchetti binari grazie al suo meccanismo di dichiarazione, nei campi di controllo dei file, delle dipendenze binarie basate su versioni. Ecco una definizione molto semplificata delle dipendenze.

- "Depends" (Dipende)
 - Dichiarare una dipendenza assoluta e tutti i pacchetti elencati in questo campo devono essere installati insieme a quello scelto o prima.
 - "Pre-Depends" (Pre-dipende)
 - È come "Depends", tranne che richiede l'installazione completa dei pacchetti elencati in anticipo.
 - "Recommends" (Raccomanda)
 - Dichiarare una dipendenza forte, ma non assoluta. La maggior parte degli utenti non vorrà il pacchetto in esame a meno che tutti i pacchetti elencati non siano installati.
 - "Suggests" (Consiglia)
 - Dichiarare una dipendenza debole. La maggior parte degli utenti del pacchetto potrebbe trarre vantaggio dall'installazione dei pacchetti elencati in questo campo, ma può ottenere una funzionalità adeguata senza di essi.
 - "Enhances" (Migliora)
 - Dichiarare una dipendenza debole come "Suggests" ma in verso opposto.
 - "Breaks" (Rompe)
 - Dichiarare un'incompatibilità del pacchetto di solito con qualche specifica versione. Generalmente la soluzione è di aggiornare tutti i pacchetti elencati in questo campo.
 - "Conflicts" (Va in conflitto)
 - Dichiarare una incompatibilità assoluta. Tutti i pacchetti elencati in questo campo devono essere rimossi per installare il pacchetto in esame.
 - "Replaces" (Sostituisce)
 - Viene dichiarata quando i file installati dal pacchetto in esame sostituiscono i file nei pacchetti elencati.
 - "Provides" (Fornisce)
-

- Viene dichiarata quando il pacchetto fornisce tutti i file e le funzionalità nei pacchetti elencati.

Nota

Notare che una buona configurazione per un pacchetto virtuale deve avere "Provides", "Conflicts" e "Replaces" simultaneamente definiti. Questo assicura che in un dato momento possa essere installato un solo pacchetto reale che fornisce il pacchetto virtuale.

La definizione ufficiale, compresa quella di dipendenza dei sorgenti, può essere trovata nel [manuale Debian Policy, Capitolo 7, Dichiarare le relazioni tra i pacchetti](#).

2.1.7 Il flusso di eventi nella gestione dei pacchetti

Ecco un riassunto semplificato del flusso di eventi nella gestione dei pacchetti con APT.

- **Aggiornamento informazioni** ("apt update", "aptitude update" o "apt-get update"):
 1. Recupera i metadati dell'archivio dall'archivio remoto
 2. Ricostruisce e aggiorna i metadati locali usati da APT
 - **Aggiornamento** («apt upgrade» e «apt full-upgrade» o «aptitude safe-upgrade» e «aptitude full-upgrade» o «apt-get upgrade» e «apt-get dist-upgrade»):
 1. Sceglie la versione candidata, che è solitamente la più recente disponibile, per tutti i pacchetti installati (per le eccezioni vedere Sezione 2.7.3)
 2. Risolve le dipendenze dei pacchetti
 3. Se la versione candidata è diversa da quella installata, scarica i pacchetti binari selezionati dall'archivio remoto
 4. Spacchetta i pacchetti binari scaricati
 5. Esegue gli script **preinst**
 6. Installa i file binari
 7. Esegue gli script **postinst**
 - **Installazione** («apt install ...», «aptitude install ...» o «apt-get install ...»):
 1. Sceglie i pacchetti elencati nella riga di comando
 2. Risolve le dipendenze dei pacchetti
 3. Scarica i pacchetti binari selezionati dall'archivio remoto
 4. Spacchetta i pacchetti binari scaricati
 5. Esegue gli script **preinst**
 6. Installa i file binari
 7. Esegue gli script **postinst**
 - **Rimozione** («apt remove ...», «aptitude remove ...» o «apt-get remove ...»):
 1. Sceglie i pacchetti elencati nella riga di comando
 2. Risolve le dipendenze dei pacchetti
 3. Esegue gli script **prerm**
 4. Rimuove i file installati **tranne** i file di configurazione
 5. Esegue gli script **postrm**
 - **Rimozione completa** («apt purge», «aptitude purge ...» o «apt-get purge ...»):
 1. Sceglie i pacchetti elencati nella riga di comando
-

2. Risolve le dipendenze dei pacchetti
3. Esegue gli script **prerm**
4. Rimuove i file installati **inclusi** i file di configurazione
5. Esegue gli script **postrm**

Sono stati intenzionalmente omessi dettagli tecnici a favore di una più chiara vista d'insieme.

2.1.8 Prima risposta a problemi di gestione dei pacchetti

Si dovrebbe leggere la bella documentazione ufficiale. Il primo documento da leggere è il file specifico per Debian, `"/usr/share/doc/".` Andrebbe consultata anche l'altra documentazione in `"/usr/share/doc/nome_pacchetto/".` Se la shell è impostata come in Sezione 1.4.2, digitare quanto segue.

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

Per informazioni dettagliate potrebbe essere necessario installare il pacchetto di documentazione corrispondente il cui nome ha il suffisso `"-doc"`.

Se si stanno avendo problemi con un pacchetto specifico, ricordarsi come prima cosa di controllare il sito del [Sistema Debian di tracciamento dei bug \(BTS\)](#).

sito web	comando
Pagina web del Sistema Debian di tracciamento dei bug (BTS)	<code>sensible-browser "http://bugs.debian.org/"</code>
Segnalazioni di bug di un pacchetto specifico	<code>sensible-browser "http://bugs.debian.org/nome_pacchetto"</code>
Segnalazione di bug corrispondente ad un numero di bug	<code>sensible-browser "http://bugs.debian.org/numero_bug"</code>

Tabella 2.5: Elenco dei siti web importanti per la risoluzione di problemi con un pacchetto specifico

Cercare con [Google](#) con parole chiave e aggiungendo `"site:debian.org"`, `"site:wiki.debian.org"`, `"site:lists.debian.org"` ecc.

Quando si invia una segnalazione di bug, usare il comando `reportbug(1)`.

2.2 Operazioni base per la gestione dei pacchetti

Le operazioni di gestione dei pacchetti basate sui repository sul sistema Debian possono essere eseguite da molti strumenti di gestione dei pacchetti basati su APT disponibili sul sistema Debian. Qui, verranno spiegati 3 strumenti base di gestione dei pacchetti: `apt`, `apt-get` / `apt-cache` e `aptitude`.

Per operazioni di gestione dei pacchetti che comportano l'installazione o l'aggiornamento dei metadati dei pacchetti, è necessario avere privilegi di root.

2.2.1 Confronti `apt` - `apt-get` / `apt-cache` - `aptitude`

Anche se `aptitude` è uno strumento interattivo molto bello ed è quello principalmente usato dall'autore, si dovrebbero conoscere alcuni avvertimenti:

- Il comando `aptitude` non è raccomandato per l'aggiornamento del sistema da un rilascio ad un altro sul sistema Debian `stable` dopo un nuovo rilascio.

- Per questo è raccomandato l'uso di «`apt full-upgrade`» o «`apt-get dist-upgrade`». Vedere il [Bug nr. 411280](#).
- Il comando `aptitude` a volte suggerisce rimozioni in massa di pacchetti per l'aggiornamento del sistema sui sistemi Debian `testing` o `unstable`.
 - Questa situazione ha spaventato molti amministratori di sistema. Niente panico.
 - Ciò sembra essere dovuto soprattutto ad una non corrispondenza tra le versioni di pacchetti da cui dipende un metapacchetto o che sono raccomandate da un metapacchetto, come `gnome-core`.
 - Ciò può essere risolto selezionando «Annulla azioni in attesa» nel menu dei comandi di `aptitude`, uscendo da `aptitude` e usando «`apt full-upgrade`».

I comandi `apt-get` e `apt-cache` sono gli strumenti di gestione dei pacchetti basati su APT più a livello **base**.

- `apt-get` e `apt-cache` offrono solo l'interfaccia utente a riga di comando.
- `apt-get` è più adatto per gli **aggiornamenti principali del sistema** tra rilasci diversi, ecc.
- `apt-get` offre un sistema **robusto** di risoluzione delle dipendenze dei pacchetti.
- `apt-get` è meno avido di risorse di sistema. Consuma meno memoria ed è più veloce durante l'esecuzione.
- `apt-cache` offre una ricerca **standard**, basata su espressioni regolari, nei nomi e nelle descrizioni dei pacchetti.
- `apt-get` e `apt-cache` possono gestire più versioni dei pacchetti usando `/etc/apt/preferences`, ma è piuttosto elaborato farlo.

Il comando `apt` è un'interfaccia a riga di comando di alto livello per la gestione dei pacchetti. Fondamentalmente è un wrapper di `apt-get`, `apt-cache` e comandi simili, originariamente pensata come interfaccia per l'utente finale e abilita in modo predefinito alcune opzioni più adatte per l'uso interattivo.

- `apt` fornisce una comoda barra di avanzamento quando si installano pacchetti usando `apt install`.
- `apt` **elimina** in modo predefinito i pacchetti `.deb` nella cache dopo l'installazione con successo dei pacchetti scaricati.

Suggerimento

È raccomandato agli utenti di usare il nuovo comando `apt(8)` per l'uso **interactive** e di usare i comandi `apt-get(8)` e `apt-cache(8)` negli script di shell.

Il comando `aptitude` è lo strumento di gestione dei pacchetti basato su APT più **versatile**.

- `aptitude` offre un'interfaccia utente testuale interattiva a schermo intero.
 - `aptitude` offre anche un'interfaccia utente a riga di comando.
 - `aptitude` è più adatto per la **gestione interattiva quotidiana dei pacchetti**, come l'ispezione dei pacchetti installati e la ricerca dei pacchetti disponibili.
 - `aptitude` è più avido di risorse di sistema. Consuma più memoria ed è più lento durante l'esecuzione.
 - `apt-cache` offre una ricerca **migliorata**, basata su espressioni regolari, su tutti i metadati dei pacchetti.
 - `aptitude` può gestire più versioni dei pacchetti senza usare `/etc/apt/preferences` e farlo è piuttosto intuitivo.
-

sintassi per apt	sintassi per aptitude	sintassi per apt-get/apt-cache	descrizione
apt update	aptitude update	apt-get update	aggiorna i metadati dell'archivio dei pacchetti
apt install pippo	aptitude install pippo	apt-get install pippo	installa la versione candidata del pacchetto "pippo" con le sue dipendenze
apt upgrade	aptitude safe-upgrade	apt-get upgrade	installa le versioni candidate dei pacchetti installati senza rimuovere altri pacchetti
apt full-upgrade	aptitude full-upgrade	apt-get dist-upgrade	installa le versioni candidate dei pacchetti installati e rimuove, se necessario, altri pacchetti
apt remove pippo	aptitude remove pippo	apt-get remove pippo	rimuove il pacchetto "pippo" lasciando i suoi file di configurazione
apt autoremove	N/D	apt-get autoremove	rimuove i pacchetti installati automaticamente che non sono più necessari
apt purge pippo	aptitude purge pippo	apt-get purge pippo	elimina completamente il pacchetto "pippo" con i suoi file di configurazione
apt clean	aptitude clean	apt-get clean	pulisce completamente il repository locale dei file dei pacchetti scaricati
apt autoclean	aptitude autoclean	apt-get autoclean	pulisce il repository locale dei file dei pacchetti scaricati che sono obsoleti
apt show pippo	aptitude show pippo	apt-cache show pippo	mostra informazioni dettagliate sul pacchetto "pippo"
apt search regex	aptitude search regex	apt-cache search regex	cerca i pacchetti che corrispondono all'espressione regolare <i>regex</i>
N/D	aptitude why regex	N/D	spiega le ragioni per cui i pacchetti che corrispondono a <i>regex</i> dovrebbero essere installati
N/D	aptitude why-not regex	N/D	spiega le ragioni per cui i pacchetti che corrispondono a <i>regex</i> non dovrebbero essere installati
N/D	aptitude search '~i!~M'	apt-mark showmanual	elenca i pacchetti installati manualmente

Tabella 2.6: Operazioni base di gestione dei pacchetti dalla riga di comando usando aptitude(8) e apt-get(8) /apt-cache(8)

2.2.2 Operazioni base per la gestione dei pacchetti dalla riga di comando

Ecco alcune operazioni base di gestione dei pacchetti dalla riga di comando usando `apt(8)`, `aptitude(8)` e `apt-get(8)` / `apt-cache(8)`.

`apt` / `apt-get` and `aptitude` can be mixed without major troubles.

«`aptitude why regex`» può mostrare ancor più informazioni se usato come «`aptitude -v why regex`». Informazioni simili possono essere ottenute con «`apt rdepends package`» o «`apt-cache rdepends package`».

Quando il comando `aptitude` è avviato in modalità a riga di comando e si trova davanti a problemi come conflitti tra pacchetti, si può passare alla modalità interattiva a tutto schermo premendo successivamente il tasto "e" al prompt.

Nota

Sebbene il comando `aptitude` sia dotato di moltissime funzionalità come il suo risolutore avanzato di conflitti tra pacchetti, questa complessità ha causato (o può ancora causare) alcune regressioni quali i [Bug nr. 411123](#), [Bug nr. 514930](#) e [Bug nr. 570377](#). In caso di dubbio, preferire i comandi `apt`, `apt-get` e `apt-cache` al comando `aptitude`.

Si possono usare opzioni per il comando immediatamente dopo "aptitude".

opzione per il comando	descrizione
-s	simula il risultato del comando
-d	scarica solamente ma non installa/aggiorna
-D	mostra delle brevi spiegazioni prima di installazioni e rimozioni automatiche

Tabella 2.7: Opzioni degne di nota per il comando `aptitude(8)`

Per maggiori informazioni vedere `aptitude(8)` e il "manuale utente di `aptitude`" in `"/usr/share/doc/aptitude/README"`.

2.2.3 Uso interattivo di aptitude

Per la gestione interattiva dei pacchetti, si avvia `aptitude` in modalità interattiva dal prompt di shell della console nel modo seguente.

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

In questo modo si aggiorna la copia locale delle informazioni sull'archivio e viene mostrato l'elenco dei pacchetti in una schermata a pieno schermo. `Aptitude` posiziona la sua configurazione in `"~/ .aptitude/config"`.

Suggerimento

Se si desidera usare la configurazione di root, invece di quella dell'utente, usare `"sudo -H aptitude ..."` invece di `"sudo aptitude ..."`, come nell'esempio precedente.

Suggerimento

`Aptitude` imposta automaticamente le **azioni pendenti** quando viene avviato in modalità interattiva. Se ciò non piace, si può reimpostarlo usando il menu: "Azioni" → "Annulla azioni in attesa".

tasto	azione associata
F10 o Ctrl-t	menu
?	mostra l' aiuto per i tasti (elenco più completo)
F10 → Aiuto → Manuale utente	mostra il Manuale utente
u	aggiorna le informazioni sull'archivio dei pacchetti
+	segna il pacchetto per l' aggiornamento o l' installazione
-	segna il pacchetto per la rimozione (mantiene i file di configurazione)
—	segna il pacchetto per l' eliminazione completa (rimuove i file di configurazione)
=	blocca il pacchetto
U	segna tutti i pacchetti aggiornabili (funziona come full-upgrade)
g	inizia a scaricare e installare i pacchetti selezionati
q	esce dalla schermata attuale e salva i cambiamenti
x	esce dalla schermata attuale e scarta i cambiamenti
Invio	visualizza le informazioni su un pacchetto
C	visualizza il registro dei cambiamenti di un pacchetto
l	cambia la limitazione dei pacchetti visualizzati
/	cerca la prima corrispondenza
\	ripete l'ultima ricerca

Tabella 2.8: Elenco delle associazioni di tasti per aptitude

2.2.4 Associazioni dei tasti per aptitude

Quelle che seguono sono le associazioni di tasti degne di nota usate, nella modalità a schermo intero, per sfogliare lo stato dei pacchetti e per impostare "azioni pianificate" su di essi.

Le espressioni regolari usate per specificare il nome di pacchetto nella riga di comando e nel prompt del menu dopo aver premuto "l" o "/" sono descritte in seguito. Le espressioni regolari di aptitude possono esplicitamente trovare corrispondenze con nomi di pacchetto usando una stringa che inizia con "~n" seguita dal nome del pacchetto.

Suggerimento

È necessario premere "U" per aggiornare tutti i pacchetti installati alla **versione candidata** nell'interfaccia interattiva. Altrimenti vengono aggiornati alla **versione candidata** solo i pacchetti selezionati e quelli da una cui particolare versione dipendono altri.

2.2.5 Viste dei pacchetti in aptitude

Nella modalità interattiva a tutto schermo di `aptitude(8)`, i pacchetti nell'elenco vengono mostrati come nell'esempio seguente.

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

Il significato degli elementi di questa riga, partendo da sinistra, è il seguente.

- Il flag di "stato attuale" (la prima lettera)
- Il flag di "azione pianificata" (la seconda lettera)
- Il flag "automatico" (la terza lettera)
- Il nome del pacchetto
- Il cambiamento in termini di uso dello spazio su disco collegato dall'"azione pianificata"
- La versione attuale del pacchetto

- La versione candidata del pacchetto

Suggerimento

L'elenco completo dei flag è fornito alla fine della schermata mostrata di **Aiuto** mostrata se si preme "?".

La **versione candidata** è scelta in base alle preferenze locali attuali (vedere `apt_preferences(5)` e Sezione 2.7.3).

Diversi tipi di viste per i pacchetti sono disponibili nel menu "Viste".

vista	descrizione della vista
Vista dei pacchetti	vedere Tabella 2.10 (vista standard)
Controlla Raccomandati	elenca i pacchetti che sono raccomandati da alcuni dei pacchetti installati, ma non sono ancora installati
Elenco unico dei pacchetti	elenca i pacchetti senza organizzarli in gruppi (per l'uso con espressioni regolari)
Consultazione per «Debtags»	elenca i pacchetti organizzati in base alle loro voci Debtags
Source Package View	list packages grouped by source packages

Tabella 2.9: Elenco delle viste di aptitude

Nota

È estremamente benvenuto ogni aiuto per [migliorare l'assegnazione di etichette Debtags ai pacchetti!](#)

La "Vista dei pacchetti" standard organizza i pacchetti un po' nello stile di `dselect` con alcune funzionalità extra.

categoria	descrizione della vista
Pacchetti aggiornabili	elenca i pacchetti organizzati in sezione → area → pacchetto
Pacchetti nuovi	" "
Pacchetti installati	" "
Pacchetti non installati	" "
Pacchetti obsoleti e creati localmente	" "
Pacchetti virtuali	elenca i pacchetti con la stessa funzione
Task	elenca i pacchetti con funzioni diverse solitamente necessarie per una stessa attività

Tabella 2.10: Organizzazione delle viste standard dei pacchetti

Suggerimento

La vista dei Task può essere usata per scegliere a uno a uno i pacchetti per le proprie attività.

2.2.6 Opzioni per i metodi di ricerca in aptitude

Aptitude offre diverse opzioni per cercare pacchetti usando il suo schema di espressioni regolari.

- Nella riga di comando della shell:
 - `"aptitude search 'regex_aptitude'"` per elencare lo stato di installazione, il nome del pacchetto ed una descrizione breve dei pacchetti che corrispondono.
-

- `aptitude show 'nome_pacchetto'` per elencare una descrizione dettagliata del pacchetto
- Nella modalità interattiva a tutto schermo:
 - `l` per limitare la vista dei pacchetti ai pacchetti che corrispondono
 - `/` per cercare un pacchetto che corrisponda
 - `\` per cercare all'indietro un pacchetto che corrisponda
 - `n` per trovare il successivo
 - `N` per trovare il successivo (all'indietro)

Suggerimento

La stringa `package_name` viene usata per trovare una corrispondenza esatta con il nome di pacchetto a meno che non venga esplicitamente fatta iniziare con `~` per indicare una espressione regolare.

2.2.7 La struttura delle espressioni regolari di aptitude

La struttura delle espressioni regolari di aptitude è quella di **ERE** (vedere Sezione 1.6.2) estese in stile mutt ed il significato delle speciali regole estese di corrispondenza, specifiche di aptitude, è il seguente.

- Il formato dell'espressione regolare è la stessa di quello **ERE** usato nei tipici strumenti per il testo in stile Unix `^`, `*`, `$`, ecc. come in `egrep(1)`, `awk(1)` e `perl(1)`.
- Il *tipo* di relazione è uno tra `depends`, `predepends`, `recommends`, `suggests`, `conflicts`, `replaces`, `provides`, e definisce la relazione tra i pacchetti.
- Il *tipo* di dipendenza predefinito è `depends`.

Suggerimento

Quando `modello_regex` è una stringa vuota, mettere `~T` immediatamente dopo il comando.

Ecco alcune scorciatoie.

- `~Ptermine` == `~Dprovides:termine`
- `~Ctermine` == `~Dconflicts:termine`
- `~W termine` == `(...|termine)`

Gli utenti che hanno familiarità con mutt imparano presto, dato che la sintassi per le espressioni si è ispirata a mutt. Vedere "SEARCHING, LIMITING, AND EXPRESSIONS" nel "Manuale utente", `/usr/share/doc/aptitude/README`.

Nota

Con la versione lenny di aptitude(8), può essere usata la nuova sintassi in **forma lunga** come `?broken` invece del vecchio equivalente in **forma breve** `~b`. Il carattere di spazio è ora considerato uno dei caratteri che terminano l'espressione regolare, oltre al carattere tilde `~`. Vedere il "Manuale utente" per la sintassi della nuova **forma lunga**.

descrizione della regola estesa di corrispondenza	struttura della espressione regolare
corrispondenza con il nome di pacchetto	<code>~nregex_nome</code>
corrispondenza con la descrizione	<code>~dregex_descrizione</code>
corrispondenza con il nome del task	<code>~tregex_task</code>
corrispondenza con il debtag	<code>~Gregex_debtag</code>
corrispondenza con il manutentore	<code>~mregex_mantainer</code>
corrispondenza con la sezione del pacchetto	<code>~sregex_sezione</code>
corrispondenza con la versione del pacchetto	<code>~Vregex_versione</code>
corrispondenza con l'archivio	<code>~A{bullseye,bookworm,sid}</code>
corrispondenza con l'origine	<code>~O{debian,...}</code>
corrispondenza con la priorità	<code>~p{extra,important,optional,required,standard}</code>
corrispondenza con pacchetti essenziali	<code>~E</code>
corrispondenza con pacchetti virtuali	<code>~V</code>
corrispondenza con pacchetti nuovi	<code>~N</code>
corrispondenza con azioni pendenti	<code>~a{install,upgrade,downgrade,remove,purge,hold,keep}</code>
corrispondenza con pacchetti installati	<code>~i</code>
corrispondenza con pacchetti installati e con l'indicatore A (pacchetti installati automaticamente)	<code>~M</code>
corrispondenza con i pacchetti installati e senza l'indicatore A (pacchetti selezionati dall'amministratore)	<code>~i!~M</code>
corrispondenza con pacchetti installati che sono aggiornabili	<code>~U</code>
corrispondenza con pacchetti rimossi ma non eliminati completamente	<code>~C</code>
corrispondenza con pacchetti rimossi, eliminati o che possono essere rimossi	<code>~g</code>
corrispondenza con pacchetti che dichiarano problemi di dipendenze	<code>~b</code>
corrispondenza con pacchetti con problemi di dipendenze di tipo <i>tipo</i>	<code>~B<tipogt;</code>
corrispondenza con pacchetti <i>modello</i> con dipendenze di tipo <i>tipo</i>	<code>~D[tipo:]modello</code>
corrispondenza con pacchetti <i>modello</i> con dipendenze non soddisfatte di tipo <i>tipo</i>	<code>~DB[tipo:]modello</code>
corrispondenza con pacchetti verso i quali il pacchetto che corrisponde a <i>modello</i> dichiara una dipendenza di tipo <i>tipo</i>	<code>~R[tipo:]modello</code>
corrispondenza con pacchetti verso i quali il pacchetto <i>termine</i> dichiara una dipendenza non soddisfatta di tipo <i>tipo</i>	<code>~RB[tipo:]modello</code>
corrispondenza con pacchetti da cui dipende un qualche altro pacchetto installato	<code>~R~i</code>
corrispondenza con pacchetti da cui non dipende alcun pacchetto installato	<code>!~R~i</code>
corrispondenza con pacchetti da cui dipende o che vengono raccomandati da qualche altro pacchetto	<code>~R~i ~Rrecommends:~i</code>
corrispondenza con pacchetto <i>modello</i> con filtro sulle versioni	<code>~S filtro modello</code>
corrispondenza con tutti i pacchetti (vero)	<code>~T</code>
corrispondenza con nessun pacchetto (falso)	<code>~F</code>

Tabella 2.11: Elenco delle regole per espressioni regolari di aptitude

2.2.8 Risoluzione delle dipendenze di aptitude

Se la voce di menu `F10` → Opzioni → Preferenze → Gestione delle dipendenze” è impostata in modo appropriato, selezionando un pacchetto in `aptitude` non si richiamano solamente i pacchetti definiti nel suo elenco `”Depends:”`, ma anche quelli nell’elenco `”Recommends:”`. Questi pacchetti installati automaticamente vengo rimossi automaticamente se non sono più necessari secondo `aptitude`.

L’opzione che controlla il comportamento di `”installazione automatica”` del comando `aptitude` può anche essere manipolata usando il comando `apt-mark(8)` dal pacchetto `apt`.

2.2.9 Registri delle attività sui pacchetti

Si può controllare lo storico delle attività sui pacchetti nei file di registro.

file	contenuto
<code>/var/log/dpkg.log</code>	Registro delle attività a livello <code>dpkg</code> per tutte le attività sui pacchetti
<code>/var/log/apt/term.log</code>	Registro delle attività APT generiche
<code>/var/log/aptitude</code>	Registro delle attività di comandi <code>aptitude</code>

Tabella 2.12: File di registro per le attività sui pacchetti

In realtà non è facile ottenere delle informazioni immediatamente comprensibili da questi file di registro. Vedere Sezione [9.3.9](#) per un metodo più facile.

2.3 Esempi di operazioni con aptitude

Ecco alcuni esempi di operazioni con `aptitude(8)`.

2.3.1 Elencare pacchetti in base alla corrispondenza del nome con espressioni regolari

Il seguente comando elenca i pacchetti i cui nomi corrispondono alle espressioni regolari specificate.

```
$ 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
```

È piuttosto utile se si desidera trovare il nome esatto di un pacchetto.

2.3.2 Sfogliare le corrispondenze ad una espressione regolare

L’espressione regolare `”~dipv6”` nel prompt di `”l”` nella vista `”Nuovo elenco unico dei pacchetti”`, limita la visualizzazione dei pacchetti a quelli la cui descrizione corrisponde all’espressione `”ipv6”` e permette di sfogliare interattivamente le loro informazioni.

2.3.3 Eliminare completamente i pacchetti rimossi

Si possono eliminare tutti i file di configurazione rimasti di pacchetti rimossi.

Controllare il risultato del comando seguente.

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

Se si pensa che sia giusto rimuovere i pacchetti elencati, eseguire il comando seguente.

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

Si può fare la stessa cosa in modalità interattiva per avere un controllo più dettagliato.

Fornire l'espressione regolare "~c" al prompt di "l" nella vista "Nuova vista dei pacchetti". In questo modo si limita la visualizzazione a quei pacchetti che corrispondono all'espressione regolare, cioè "rimossi ma non eliminati completamente". Tutte le corrispondenze trovate possono essere mostrate premendo "[" nelle intestazioni di primo livello.

Premere poi "_" nell'intestazione di primo livello, come "Pacchetti non installati". In questo modo vengono selezionati per l'eliminazione completa solo quei pacchetti contenuti sotto l'intestazione che hanno fatto corrispondenza con l'espressione regolare. Si possono escludere alcuni pacchetti in modo interattivo dall'eliminazione premendo "=" per ciascuno di essi.

Questa tecnica è piuttosto comoda e funziona per molti altri tasti di comando.

2.3.4 Mettere ordine nello stato di installazione automatico/non automatico

Ecco come faccio per mettere ordine nello stato di installazione automatico/non automatico per i pacchetti (dopo aver usato un installatore che non sia aptitude, ecc.).

1. Avviare da root `aptitude` in modalità interattiva.
2. Digitare "u", "U", "f" e "g" per aggiornare l'elenco dei pacchetti e i pacchetti stessi.
3. Digitare "l" per impostare la limitazione della visualizzazione con "~i(~R~i|~Rrecommends:~i)" e digitare "M" su "Pacchetti installati" per marcarli come installati automaticamente.
4. Digitare "l" per impostare la limitazione della visualizzazione dei pacchetti con "~prequired|~pimportant|~pstandard" e digitare "m" su "Pacchetti installati" per marcarli come installati manualmente.
5. Digitare "l" per impostare la limitazione della visualizzazione dei pacchetti con "~i!~M" e rimuovere i pacchetti inutilizzati digitando "-" su ciascuno di essi dopo averli visualizzati digitando "[" su "Pacchetti installati".
6. Digitare "l" per impostare la limitazione della visualizzazione dei pacchetti con "~i"; poi digitare "m" su "Task" per marcare quei pacchetti come installati manualmente.
7. Uscire da `aptitude`.
8. Avviare `apt-get -s autoremove|less` da root per controllare quelli che non sono usati.
9. Riavviare `aptitude` in modalità interattiva e segnare i pacchetti necessari con "m".
10. Riavviare `apt-get -s autoremove|less` da root per controllare che in REMOVED siano contenuti soltanto i pacchetti attesi.
11. Avviare `apt-get autoremove|less` da root per rimuovere automaticamente i pacchetti non utilizzati.

L'azione "m" sopra "Task" è opzionale e server per prevenire situazioni di rimozioni di massa di pacchetti in futuro.

2.3.5 Aggiornamento di tutto il sistema

Nota

Quando si passa ad un nuovo rilascio, ecc. si dovrebbe considerare una installazione pulita di un nuovo sistema anche se è possibile aggiornare Debian nel modo descritto in seguito. Ciò dà la possibilità di rimuovere la spazzatura raccolta nel tempo e fornisce la migliore combinazione dei pacchetti più nuovi. Naturalmente si dovrebbe fare un backup completo del sistema in un posto sicuro (vedere Sezione [10.2](#)) prima di farlo. Raccomando di creare una configurazione con due sistemi avviabili usando due diverse partizioni per avere una transizione il più lineare possibile.

Si può fare l'aggiornamento completo del sistema ad un nuovo rilascio cambiando il contenuto del file `«/etc/apt/sources.list»` facendolo puntare al nuovo rilascio ed eseguendo il comando `«apt update; apt dist-upgrade»`.

Per aggiornare da `stable` a `testing` o `unstable`, sostituire `""bullseye"` nel file `"/etc/apt/sources.list"` di esempio descritto in Sezione 2.1.4 con `"bookworm"` o `"sid"`.

In realtà si possono dover affrontare alcune complicazioni a causa di problemi di transizione dei pacchetti, per lo più a causa di dipendenze. Più è grande la differenza nell'aggiornamento, più è probabile dover affrontare problemi più grossi. Per la transizione dal vecchio rilascio `stable` al nuovo `stable` si possono leggere le nuove [Note di rilascio](#) e seguire esattamente la procedura che vi è descritta per minimizzare i problemi.

Quando si decide di passare da `stable` a `testing` prima del suo rilascio formale non ci sono [Note di rilascio](#) che possano aiutare. La differenza tra `stable` e `testing` potrebbe essere diventata piuttosto grande dopo il precedente rilascio `stable` e potrebbe rendere l'aggiornamento complicato.

SI dovrebbero prendere misure precauzionali per l'aggiornamento completo raccogliendo le informazioni più aggiornate da mailing list ed usando il buon senso.

1. Leggere le "Note di rilascio" precedenti.
2. Fare il backup dell'intero sistema (specialmente dei dati e della configurazione).
3. Tenere a portata di mano supporti avviabili in caso di problemi con il bootloader.
4. Informare con un buon anticipo gli utenti nel sistema.
5. Registrare l'attività di aggiornamento con `script(1)`.
6. Applicare l'opzione `"unmarkauto"` ai pacchetti richiesti, ad esempio `"aptitude unmarkauto vim`, per prevenire la loro rimozione.
7. Minimizzare i pacchetti installati per ridurre le probabilità di conflitti tra i pacchetti, ad esempio rimuovere i pacchetti dell'attività desktop.
8. Rimuovere il file `"/etc/apt/preferences"` (disabilitare apt-pinning).
9. Cercare di aggiornare per gradi: `oldstable` → `stable` → `testing` → `unstable`.
10. Aggiornare il file `"/etc/apt/sources.list"` per farlo puntare solamente al nuovo archivio ed eseguire `"aptitude update"`.
11. Installare, opzionalmente i nuovi pacchetti partendo dai **pacchetti fondamentali**, ad esempio `"aptitude install perl"`.
12. Eseguire il comando `"apt-get -s dist-upgrade"` per avere un'idea dell'impatto dell'aggiornamento.
13. Eseguire da ultimo il comando `"apt-get dist-upgrade"`.

**Attenzione**

Non è saggio saltare un rilascio principale Debian quando si aggiorna da un rilascio `stable` ad un altro.

**Attenzione**

Nelle "Note di rilascio" precedenti GCC, il kernel Linux, `initrd-tools`, Glibc, Perl, APT tool chain, ecc. hanno richiesto un po' di attenzione particolare per gli aggiornamenti dell'intero sistema.

Per gli aggiornamenti quotidiani in `unstable`, vedere Sezione 2.4.3.

2.4 Operazioni avanzate per la gestione dei pacchetti

2.4.1 Operazioni avanzate per la gestione dei pacchetti dalla riga di comando

Ecco un elenco di altre operazioni di gestione dei pacchetti per i quali `aptitude` è di livello troppo alto o manca delle funzionalità richieste.

Nota

Per un pacchetto con la funzionalità `multi-arch`, può essere necessario specificare l'architettura per alcuni comandi. Per esempio, usare `dpkg -L libglib2.0-0:amd64` per elencare il contenuto del pacchetto `libglib2.0-0` per l'architettura `amd64`.



Attenzione

Gli strumenti per i pacchetti a più basso livello, come `dpkg -i ...` e `debi ...` dovrebbero essere usati con molta cura dall'amministratore di sistema. Non si prendono cura in modo automatico delle dipendenze richieste dai pacchetti. Le opzioni `--force-all` e simili, per la riga di comando di `dpkg` (vedere `dpkg(1)`), sono pensate per essere usate solamente da utenti esperti. Usarle senza comprenderne pienamente l'effetto potrebbe portare tutto il sistema in uno stato problematico.

Notare quanto segue.

- Tutti i comandi di configurazione ed installazioni di sistema devono essere eseguiti da `root`.
- A differenza di `aptitude` che usa espressioni regolari (vedere Sezione 1.6.2), gli altri comandi di gestione dei pacchetti usano espansioni di modelli in stile shell (vedere Sezione 1.5.6).
- Per `apt-file(1)`, fornito dal pacchetto `apt-file`, è necessario eseguire prima `apt-file update`.
- `configure-debian(8)`, fornito dal pacchetto `configure-debian` esegue `dpkg-reconfigure(8)` come suo backend.
- `dpkg-reconfigure(8)` esegue gli script dei pacchetti usando `debconf(1)` come suo backend.
- I comandi `apt-get build-dep`, `apt-get source` e `apt-cache showsrc` richiedono la voce `deb-src` in `/etc/apt/sources.list`.
- `dget(1)`, `debuild(1)` e `debi(1)` richiedono il pacchetto `devscripts`.
- Vedere la procedura per (ri)creare pacchetti usando `apt-get source` in Sezione 2.7.13.
- Il comando `make-kpkg` richiede il pacchetto `kernel-package` (vedere Sezione 9.10).
- Vedere Sezione 12.9 per informazioni generali sulla creazione di pacchetti.

2.4.2 Verifica dei pacchetti installati

L'installazione di `debsums` permette la verifica dei file di pacchetto installati in base ai valori delle somme di controllo MD5sum nei file `/var/lib/dpkg/info/*.md5sums` con `debsums(1)`. Per il funzionamento di MD5sum vedere Sezione 10.3.5.

Nota

Dato che il database MD5sum può essere manomesso da un intruso, `debsums` ha un'utilità limitata come strumento di sicurezza. È adatto solo per controllare modifiche fatte dall'amministratore o danni dovuti ad errori dei supporti.

comando	azione
<code>COLUMNS=120 dpkg -l modello_nome_pacchetto</code>	elenca lo stato di un pacchetto installato per una segnalazione di bug
<code>dpkg -L nome_pacchetto</code>	elenca il contenuto di un pacchetto installato
<code>dpkg -L nome_pacchetto egrep '/usr/share/man/man.*/.+'</code>	elenca le pagine man di un pacchetto installato
<code>dpkg -S modello_nome_file</code>	elenca i pacchetti installati che uno un file con un nome che corrisponde al modello
<code>apt-file search modello_nome_file</code>	elenca i pacchetti nell'archivio che hanno un file con un nome che corrisponde al modello
<code>apt-file list modello_nome_pacchetto</code>	elenca il contenuto dei pacchetti nell'archivio che corrispondono al modello
<code>dpkg-reconfigure nome_pacchetto</code>	riconfigura il pacchetto con esattamente quel nome
<code>dpkg-reconfigure -plow package_name</code>	riconfigura il pacchetto con esattamente quel nome ponendo le domande più dettagliate
<code>configure-debian</code>	riconfigura i pacchetti da un'interfaccia a tutto schermo a menu
<code>dpkg --audit</code>	controlla il sistema alla ricerca di pacchetti parzialmente installati
<code>dpkg --configure -a</code>	configura tutti i pacchetti parzialmente installati
<code>apt-cache policy nome_pacchetto_binario</code>	mostra la versione disponibile, la priorità e le informazioni nell'archivio di un pacchetto binario
<code>apt-cache madison <nome pacchetto></code>	mostra la versione disponibile e le informazioni nell'archivio di un pacchetto
<code>apt-cache showsrc nome_pacchetto_binario</code>	mostra informazioni sul pacchetto sorgente di un pacchetto binario
<code>apt-get build-dep nome_pacchetto</code>	installa i pacchetti richiesti per compilare un pacchetto
<code>aptitude build-dep nome_pacchetto</code>	installa i pacchetti richiesti per compilare un pacchetto
<code>apt-get source nome_pacchetto</code>	scarica un sorgente (da un archivio standard)
<code>dget URL for dsc file</code>	scarica un pacchetto sorgente (da un altro archivio)
<code>dpkg-source -x nome_pacchetto_versione-versione_debian deb</code>	crea un albero dei sorgenti a partire da un insieme di pacchetti sorgenti ("*.deb", "*.dsc", "*.tar.gz" e "*.diff.gz")
<code>debuild binary</code>	compila pacchetti da un albero dei sorgenti locale
<code>make-kpkg immagine_kernel</code>	compila un pacchetto kernel a partire da un albero dei sorgenti del kernel
<code>make-kpkg --initrd immagine_kernel</code>	compila un pacchetto kernel con initramfs abilitato, a partire da un albero dei sorgenti
<code>dpkg -i nome_pacchetto_versione-versione_debian_arch.deb</code>	installa un pacchetto locale nel sistema
<code>apt install /percorso/di/nomefile_pacchetto.deb</code>	installa un pacchetto locale nel sistema, cercando al contempo di risolvere automaticamente le dipendenze
<code>debi nome_pacchetto_versione-versione_debian_arch.dsc</code>	installa pacchetti locali nel sistema
<code>dpkg --get-selections '*' >selection.txt</code>	salva le informazioni di stato sui pacchetti selezionati a livello di dpkg
<code>dpkg --set-selections <selection.txt</code>	imposta le informazioni di stato sui pacchetti selezionati a livello di dpkg
<code>echo nome_pacchetto hold dpkg --set-selections</code>	imposta lo stato di selezione di un pacchetto a livello di dpkg a hold (equivalente a "aptitude holdnome_pacchetto")

Tabella 2.13: Elenco delle operazioni avanzate per la gestione dei pacchetti

2.4.3 Salvaguardarsi da problemi coi pacchetti

Molti utenti preferiscono usare il rilascio **unstable** del sistema Debian per avere le nuove funzionalità ed i nuovi pacchetti. Questo rende il sistema più esposto a bug critici sui pacchetti.

L'installazione del pacchetto `apt-listbugs` salvaguarda il sistema dai bug critici controllando automaticamente, al momento degli aggiornamenti fatti con il sistema APT, il BTS Debian alla ricerca dei bug critici.

L'installazione del pacchetto `apt-listchanges` fornisce importanti notizie contenute in `NEWS.Debian` quando si fanno aggiornamenti con il sistema APT.

2.4.4 Cercare tra i metadati dei pacchetti

Sebbene visitare il sito Debian <https://packages.debian.org/> faciliti oggi giorno la ricerca nei metadati dei pacchetti, ci sono metodi più tradizionali per farlo.

I comandi `grep-dctrl(1)`, `grep-status(1)` e `grep-available(1)` possono essere usati per cercare in qualsiasi file che sia nel formato generico dei file di controllo dei pacchetti di Debian.

`dpkg -S modello_nome_file` può essere usato per cercare i pacchetti che contengono al loro interno file installati da `dpkg` il cui nome corrisponde al modello. Ma non trova i file creati dagli script dei manutentori.

Se è necessario fare ricerche più elaborate sui metadati dei pacchetti, è necessario eseguire il comando `grep -e modello_regex *` nella directory `/var/lib/dpkg/info/`. In questo modo si cercano le parole indicate negli script e nei testi con le domande di installazione dei pacchetti.

Se si desiderano vedere in modo ricorsivo le dipendenze dei pacchetti, si dovrebbe usare `apt-rdepends(8)`.

2.5 Aspetti tecnici della gestione dei pacchetti in Debian

Verranno ora trattati alcuni aspetti tecnici interni del funzionamento del sistema di gestione dei pacchetti Debian. Queste informazioni dovrebbero aiutare a trovare soluzioni proprie ai problemi con i pacchetti.

2.5.1 Metadati degli archivi

I file con i metadati per ciascuna distribuzione sono archiviati in `dist/codename` su ciascun sito mirror di Debian, ad esempio `http://deb.debian.org/debian/`. La struttura del suo archivio può essere navigata con il browser web. Ci sono 6 tipi di metadati.

Negli archivi recenti questi metadati sono memorizzati come file compressi e file di differenze per ridurre il traffico di rete.

2.5.2 File "Release" nella directory principale ed autenticità

Suggerimento

Il file "Release" nella directory di livello più alto è usato per firmare gli archivi nel sistema **secure APT**.

Ogni suite dell'archivio Debian ha un file "Release" nella sua directory di livello più alto, ad esempio `http://deb.debian.org/` di questo tipo.

```
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
```

file	posizione	contenuto
Release	directory della distribuzione di più alto livello	descrizione dell'archivio e informazioni sull'integrità
Release.gpg	directory della distribuzione di più alto livello	file di firma per il file "Release" firmato con la chiave dell'archivio
Contents-architettura	directory della distribuzione di più alto livello	elenca tutti i file per tutti i pacchetti nell'archivio relativo
Release	directory più elevata per ciascuna combinazione distribuzione/area/architettura	descrizione dell'archivio usata per le regole di apt_preferences(5)
Packages	directory più elevata per ciascuna combinazione distribuzione/area/architettura-binaria	dati debian/control concatenati dei pacchetti binari
Sources	directory più elevata per ciascuna combinazione distribuzione/area/source	dati debian/control concatenati dei pacchetti sorgenti

Tabella 2.14: Il contenuto dei metadati dell'archivio Debian

```

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

Qui si può scoprire il motivo per cui in questa guida si è scelto di usare "suite" e "codename" (nome in codice) in Sezione 2.1.4. Il termine "distribuzione" è usato per riferirsi sia a "suite" sia a "codename". Tutti i nomi delle "aree" dell'archivio offerte da un archivio sono elencate in "Componente".

L'integrità del file "Release" nella directory base viene verificata usando l'infrastruttura crittografica chiamata [secure apt](#).

- Il file "Release.gpg" con la firma crittografica viene creato a partire dal file "Release" nella directory base e dalla chiave segreta dell'archivio Debian.
- La chiave pubblica dell'archivio Debian può essere inserita in `/etc/apt/trusted.gpg`
 - automaticamente installando il portachiavi con il file `base-files` più recente o
 - manualmente dagli strumenti `gpg` o `apt-key` con [la più recente chiave pubblica dell'archivio pubblicata su ftp-master.debian.org](#).
- Il sistema **secure APT** verifica crittograficamente l'integrità del file "Release" di più alto livello scaricato con questo file "Release.gpg" e la chiave pubblica dell'archivio Debian in `/etc/apt/trusted.gpg`.

L'integrità di tutti i file "Packages" e "Sources" viene verificata usando i valori MD5sum nel file "Release" nella rispettiva directory base. L'integrità di tutti i file di pacchetto viene verificata usando i valori MD5sum nei file "Packages" e "Sources". Vedere `debsums(1)` e Sezione 2.4.2.

Dato che la verifica crittografica della firma è un processo molto più dispendioso in termini di uso della CPU rispetto al calcolo di valori MD5sum, l'uso di valori MD5sum per ciascun pacchetto mentre viene usata la firma crittografica per il file "Release" principale, fornisce [una buona sicurezza con buone prestazioni](#) (vedere Sezione 10.3).

2.5.3 File "Release" a livello di archivio

Suggerimento

I file "Release" a livello di archivio sono usati per le regole di `apt_preferences(5)`.

Ci sono file "Release" a livello di archivio per tutte le posizioni nell'archivio specificate dalla riga "deb" in `/etc/apt/sources.list`, come `http://ftp.us.debian.org/debian/dists/unstable/main/binary-amd64/Release` o `http://deb.debian.org/debian/dists/unstable/main/binary-amd64/Release`, nell'esempio seguente.

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



Attenzione

Nell'[archivio Debian](#), per la voce «Archive:» vengono usati i nomi di suite («stable», «testing», «unstable», ...), mentre vengono usati i nomi in codice («trusty», «xenial», «artful», ...) nell'[archivio Ubuntu](#).

Per alcuni archivi, come `experimental` e `bullseye-backports`, che contengono pacchetti che non dovrebbero essere installati automaticamente, c'è una riga aggiuntiva, come `http://deb.debian.org/debian/dists/experimental/main/binary-amd64/Release`, nell'esempio seguente.

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

Notare che per gli archivi normali senza la riga `NotAutomatic: yes`, il valore predefinito per la priorità di Pin è 500, mentre per gli archivi speciali con `NotAutomatic: yes`, il valore predefinito per la priorità di Pin è 1 (vedere `apt_preferences(5)` e Sezione [2.7.3](#)).

2.5.4 Recuperare i metadati per un pacchetto

Quando si usano strumenti APT, quali `aptitude`, `apt-get`, `synaptic`, `apt-file`, `auto-apt`, ... è necessario aggiornare le copie locali dei metadati contenenti le informazioni sull'archivio Debian. Queste copie locali hanno i seguenti nomi di file che corrispondono ai nomi di distribuzione, area e architettura specificati in `/etc/apt/sources.list` (vedere Sezione [2.1.4](#)).

- `/var/lib/apt/lists/deb.debian.org_debian_dists_distribuzione_Release`
 - `/var/lib/apt/lists/deb.debian.org_debian_dists_distribuzione_Release.gpg`
 - `/var/lib/apt/lists/deb.debian.org_debian_dists_distribuzione_area_binary-architettura_Packages`
 - `/var/lib/apt/lists/deb.debian.org_debian_dists_distribuzione_area_source_Sources`
 - `/var/cache/apt/apt-file/deb.debian.org_debian_dists_distribuzione_Contents-architettura` (per `apt-file`)
-

I primi quattro tipi di file sono condivisi da tutti i comandi APT pertinenti e vengono aggiornati dalla riga di comando con "apt-get update" e "aptitude update". I metadati "Packages" sono aggiornati se esiste la riga "deb" in "/etc/apt/sources.list". I metadati "Sources" sono aggiornati se esiste la riga "deb-src" in "/etc/apt/sources.list".

I metadati "Packages" e "Sources" contengono la voce "Filename:" che punta alla posizione dei file dei pacchetti binari e sorgenti. Attualmente questi pacchetti sono posizionati nell'albero di directory "pool/" per una transizione migliore tra i rilasci.

Le copie locali dei metadati "Packages" possono essere ricercate interattivamente con l'aiuto di aptitude. Il comando specializzato di ricerca `grep-dctrl(1)` può cercare nelle copie locali dei metadati "Packages" e "Sources".

La copia locale dei metadati "Contents-architettura" può essere aggiornata con "apt-file update" e la sua posizione è diversa da quella degli altri quattro file. Vedere `apt-file(1)`. (auto-apt usa in modo predefinito una posizione differente per la copia locale di "Contents-architettura.gz".)

2.5.5 Lo stato dei pacchetti per APT

In aggiunta ai metadati recuperati da remoto, lo strumento APT, a partire da lenny, memorizza le informazioni generate localmente sullo stato delle installazioni in "/var/lib/apt/extended_states" che è usato da tutti gli strumenti APT per tenere traccia dei pacchetti installati automaticamente.

2.5.6 Lo stato dei pacchetti per aptitude

In aggiunta ai metadati recuperati da remoto, il comando aptitude memorizza le informazioni generate localmente sullo stato delle installazioni in "/var/lib/aptitude/pkgstates" che è usato solo da esso.

2.5.7 Copie locali dei pacchetti scaricati

Tutti i pacchetti scaricati da remoto attraverso il meccanismo APT vengono salvati nella directory "/var/cache/apt/archives" fino a che non vengono cancellati.

Questa politica di pulizia dei file della cache per aptitude può essere impostata in "Opzioni" → "Preferenze" e può essere forzata usando la sua voce di menu "Pulisci la cache dei pacchetti" o "Cancella i file vecchi" in "Azioni".

2.5.8 Nomi dei file di pacchetto di Debian

I file di pacchetto Debian hanno nomi con una particolare struttura.

tipo di pacchetto	struttura del nome
Il pacchetto binario (alias deb)	<i>nome-pacchetto_<versione-originale> gt;-versione-debian_architettura.deb</i>
Il pacchetto binario per l'installatore Debian (alias udeb)	<i>nome-pacchetto_versione-originale-versione-debian_architettura.udeb</i>
Il pacchetto sorgente (sorgenti originali)	<i>nome-pacchetto_versione-originale-versione-debian.orig</i>
Il pacchetto sorgente 1.0 (modifiche Debian)	<i>nome-pacchetto_versione-originale-versione-debian.diff</i>
Il pacchetto sorgente 3.0 (quilt) (modifiche Debian)	<i>nome-pacchetto_versione-originale-versione-debian.debian</i>
Il pacchetto sorgente (descrizione)	<i>nome-pacchetto_versione-originale-versione-debian.dsc</i>

Tabella 2.15: La struttura dei nomi dei pacchetti Debian

Suggerimento

Sono qui descritti solamente i formati base dei pacchetti sorgenti. Per ulteriori informazioni vedere `dpkg-source(1)`.

nome porzione	usable characters (ERE regex)	presenza
<i>nome-pacchetto</i>	<code>[a-z0-9] [-a-z0-9.+] +</code>	richiesta
<i>epoca:</i>	<code>[0-9] +:</code>	opzionale
<i>versione-originale</i>	<code>[-a-zA-Z0-9.+:] +</code>	richiesta
<i>versione.debian</i>	<code>[a-zA-Z0-9.+~] +</code>	opzionale

Tabella 2.16: I caratteri utilizzabili all'interno di ciascuna porzione del nome dei pacchetti Debian

Nota

Si può controllare l'ordine delle versioni dei pacchetti con `dpkg(1)`, per esempio con `"dpkg --compare-versions 7.0 gt 7.~pre1 ; echo $?"`.

Nota

L'[installatore Debian \(d-i\)](#) usa come estensione per i suoi pacchetti binari `udeb` invece della normale `deb`. Un pacchetto `udeb` è un pacchetto `deb` ridotto al minimo, con rimossi alcuni contenuti non essenziali come la documentazione, per risparmiare spazio seguendo meno rigorosamente i requisiti richiesti dalle norme per i pacchetti. Entrambi i pacchetti `deb` e `udeb` condividono la stessa struttura di pacchetto. La "u" sta per micro.

2.5.9 Il comando dpkg

`dpkg(1)` è lo strumento di più basso livello per la gestione dei pacchetti Debian. È molto potente e deve essere usato con cautela. Quando installa un pacchetto chiamato "*nome_pacchetto*", `dpkg` fa nell'ordine elencato le seguenti operazioni.

1. Spacchetta il file `deb` (equivalente di `"ar -x"`)
2. Esegue "*nome_pacchetto.preinst*" usando `debconf(1)`
3. Installa il contenuto del pacchetto nel sistema (equivalente di `"tar -x"`)
4. Esegue "*nome_pacchetto.postinst*" usando `debconf(1)`

Il sistema `debconf` fornisce un'interazione standard con l'utente con supporto per internazionalizzazione e localizzazione (Capitolo 8).

Il file "`status`" è anche usato da altri strumenti come `dpkg(1)`: `"dselect update"` e `"apt-get -u dselect-upgrade"`.

Il comando di ricerca specializzato `grep-dctrl(1)` può ricercare nelle copie locali dei metadati "`status`" e "`available`".

Suggerimento

Nell'ambiente dell'[installatore Debian](#), viene usato il comando `udpkg` per aprire i pacchetti `udeb`. Il comando `udpkg` è una versione ridotta al minimo del comando `dpkg`.

2.5.10 Il comando update-alternatives

Il sistema Debian ha un meccanismo per installare senza problemi programmi che si sovrappongono nelle loro funzioni usando `update-alternatives(1)`. Per esempio, si può far sì che il comando `vi` esegua `vim` quando si sono installati entrambi i pacchetti `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
...
```

file	descrizione del contenuto
/var/lib/dpkg/info/nome_pacchetto.conf	file di configurazione (modificabile dall'utente)
/var/lib/dpkg/info/nome_pacchetto.list	elenco dei file e delle directory installate dal pacchetto
/var/lib/dpkg/info/nome_pacchetto.md5sums	elenco dei valori degli hash MD5 per i file installati dal pacchetto
/var/lib/dpkg/info/nome_pacchetto.preinst	script del pacchetto da eseguire prima dell'installazione del pacchetto
/var/lib/dpkg/info/nome_pacchetto.postinst	script del pacchetto da eseguire dopo l'installazione del pacchetto
/var/lib/dpkg/info/nome_pacchetto.prerm	script del pacchetto da eseguire prima della rimozione del pacchetto
/var/lib/dpkg/info/nome_pacchetto.postrm	script del pacchetto da eseguire dopo la rimozione del pacchetto
/var/lib/dpkg/info/nome_pacchetto.config	script per il sistema debconf
/var/lib/dpkg/alternatives/nome_pacchetto	informazioni sulle alternative usate dal comando update-alternatives
/var/lib/dpkg/available	informazioni sulla disponibilità per tutti i pacchetti
/var/lib/dpkg/diversions	le informazioni sulle deviazioni usate da dpkg(1) e impostate da dpkg-divert(8)
/var/lib/dpkg/statoverride	le informazioni per la sovrascrittura dello stato usate da dpkg(1) e impostate da dpkg-statoverride(8)
/var/lib/dpkg/status	le informazioni sullo stato per tutti i pacchetti
/var/lib/dpkg/status-old	la prima generazione di backup del file "/var/lib/dpkg/status"
/var/backups/dpkg.status*	la seconda generazione e le generazioni precedenti di backup del file "/var/lib/dpkg/status"

Tabella 2.17: I file degni di nota creati da dpkg

```
$ 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
```

Il sistema delle alternative Debian mantiene le proprie selezioni in forma di collegamenti simbolici in `"/etc/alternatives/"`. Il procedimento di selezione usa i file corrispondenti in `"/var/lib/dpkg/alternatives/"`.

2.5.11 Il comando dpkg-statoverride

La **sovrascrittura dello stato** fornita dal comando `dpkg-statoverride(8)` è un modo per dire a `dpkg(1)` di usare un proprietario o una modalità diversa per un **file** quando viene installato un pacchetto. Se viene specificato `"- -update"` e il file esiste già, viene immediatamente impostato con il proprietario e la modalità nuovi.



Attenzione

La modifica diretta del proprietario o della modalità, usando `chmod` o `chown`, di un **file** che appartiene ad un pacchetto da parte dell'amministratore del sistema viene vanificata dal successivo aggiornamento del pacchetto.

Nota

È stata usata la parola **file**, ma in realtà si può trattare di un qualsiasi oggetto nel file system gestito da `dpkg`, inclusi directory, device, ecc.

2.5.12 Il comando `dpkg-divert`

Le **deviazioni** per i file fornite dal comando `dpkg-divert(8)` sono un modo per forzare `dpkg(1)` a non installare un file nella sua posizione predefinita, ma in una posizione **deviata**. L'uso di `dpkg-divert` è pensato per gli script di manutenzione dei pacchetti. Il suo uso in maniera superficiale da parte dell'amministratore di sistema è deprecato.

2.6 Ripristino da un sistema con problemi

Quando si usa il sistema `unstable`, l'amministratore deve saper ripristinare le cose da una situazione problematica di gestione dei pacchetti.



Attenzione

Alcuni dei metodi descritti comportano azioni ad alto rischio. Ci si consideri avvertiti!

2.6.1 Incompatibilità con vecchie configurazioni utente

Se un programma con interfaccia grafica per il desktop presenta problemi di instabilità dopo un importante aggiornamento della versione originale, si dovrebbero sospettare interferenze con vecchi file locali di configurazione creati da esso. Se è stabile per account utente creati da zero, questa ipotesi è confermata. (Questo è un bug nella creazione del pacchetto e viene solitamente evitato dal curatore del pacchetto.)

Per ripristinare la stabilità, si dovrebbero spostare i corrispondenti file locali di configurazione e riavviare il programma GUI. Potrebbe essere necessario leggere il contenuto dei vecchi file di configurazione per ripristinare successivamente le informazioni di configurazione. (Non cancellarli troppo di fretta.)

2.6.2 Pacchetti diversi con file sovrapposti

I sistemi di gestione dei pacchetti a livello dell'archivio, come `aptitude(8)` o `apt-get(1)` non provano nemmeno ad installare pacchetti con file sovrapposti, basandosi sulle dipendenze dei pacchetti (vedere Sezione 2.1.6).

Errori dei manutentori dei pacchetti o l'uso di fonti degli archivi mescolate in modo incoerente (vedere Sezione 2.7.2) dall'amministratore di sistema possono creare una situazione con dipendenze dei pacchetti non correttamente definite. Quando, in una situazione di questo tipo, si installa un pacchetto con file sovrapposti usando `aptitude(8)` o `apt-get(1)`, `dpkg(1)` che spacchetta i pacchetti si assicura di restituire un errore al programma che lo ha invocato senza sovrascrivere file esistenti.



Attenzione

L'uso di pacchetti di terze parti introduce significativi rischi per il sistema attraverso gli script dei manutentori che sono eseguiti con privilegi di root e che possono fare qualsiasi cosa nel sistema. Il comando `dpkg(1)` protegge solo contro la sovrascrittura in fase di spaccettamento.

Si possono risolvere installazioni problematiche di questo tipo rimuovendo prima il vecchio pacchetto che crea problemi: *vecchio-pacchetto*

```
$ sudo dpkg -P vecchio-pacchetto
```

2.6.3 Risolvere problemi negli script di pacchetto

Quando un comando nello script di un pacchetto restituisce un errore per una qualche ragione e lo script finisce con un errore, il sistema di gestione dei pacchetti cancella l'azione e finisce con pacchetti parzialmente installati. Quando un pacchetto contiene bug nei suoi script di rimozione, il pacchetto può diventare impossibile da rimuovere e fonte di problemi.

Per i problemi di script di pacchetto di *"nome_pacchetto"*, si dovrebbe guardare negli script di pacchetto seguenti.

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

Modificare dall'account root lo script di pacchetto che causa problemi usando le tecniche seguenti.

- disabilitare la riga che causa problemi facendola precedere da *"#"*
- forzare la restituzione di esecuzione con successo aggiungendo in fondo alla riga che causa problemi *"| | true"*

Configurare tutti i pacchetti parzialmente installati con il comando seguente.

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

2.6.4 Ripristinare con il comando dpkg

Dato che `dpkg` è uno strumento per pacchetti a livello molto basso, può funzionare anche in situazioni molto brutte come un sistema non avviabile senza una connessione di rete. Assumiamo, per esempio, che il pacchetto `pippo` sia difettoso e debba essere sostituito.

Si possono trovare copie di versioni più vecchie del pacchetto `pippo`, libere dal bug, nella directory della cache dei pacchetti: `«/var/cache/apt/archives/»`. (Se non ci sono, se ne può scaricare una dall'archivio di <https://snapshot.debian.org/> o copiarla dalla cache dei pacchetti di una macchina funzionante.)

Se si può avviare il sistema, lo si può installare con il comando seguente.

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

Suggerimento

Se il danno al sistema è piccolo, si può in alternativa retrocedere tutto il sistema ad una versione precedente come in Sezione 2.7.10, usando il sistema di più alto livello APT.

Se il sistema non è avviabile dal disco fisso, si dovrebbe cercare un altro modo per fare l'avvio.

1. Avviare il sistema usando il CD dell'installatore Debian in modalità ripristino.
2. Montare il sistema non avviabile sul disco fisso in *"/destinazione"*.
3. Installare una versione più vecchia del pacchetto `pippo` con il comando seguente.

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

Questo esempio funziona persino se il comando `dpkg` sul disco fisso è danneggiato.

Suggerimento

Per ripristinare il sistema danneggiato, può essere usato in modo simile qualsiasi sistema GNU/Linux avviato da un altro sistema sul disco fisso, CD live GNU/Linux, chiavetta USB avviabile o avvio di rete.

Se il tentativo di installare un pacchetto in questo modo fallisce a causa di qualche violazione di dipendenze e si deve necessariamente farlo come ultima spiaggia, si possono sovrascrivere le dipendenze usando `"--ignore-depends"`, `"--force-depends"` e altre opzioni di `dpkg`. Se lo si fa, è necessario in seguito fare tutti gli sforzi possibili per ripristinare le dipendenze corrette. Per i dettagli vedere `dpkg(8)`.

Nota

Se il sistema è seriamente danneggiato, se ne dovrebbe fare un backup completo in un posto sicuro (vedere Sezione 10.2) e fare un'installazione pulita. Ciò comporta meno perdita di tempo e alla fine produce un risultato migliore.

2.6.5 Ripristinare i dati sui pacchetti selezionati

Se `/var/lib/dpkg/status` diventa corrotto per una qualsiasi ragione, il sistema Debian perde i dati sulla selezione dei pacchetti e ne ha grave danno. Cercare il vecchio file `/var/lib/dpkg/status` in `/var/lib/dpkg/status-old` o `/var/backups/dpkg.status.*`.

Mantenere `/var/backups/` in una partizione separata può essere una buona idea dato che questa directory contiene molti dati importanti per il sistema.

Per danni molto seri, raccomando di fare una re-installazione pulita dopo aver fatto il backup del sistema. Anche se tutto ciò che era in `/var/` è andato perduto, è sempre possibile recuperare alcune informazioni dalle directory in `/usr/share/doc/` per guidare le nuove installazioni.

Reinstallare un sistema (desktop) minimale

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

Montare il vecchio sistema in `"percorso/del/vecchio/sistema/"`.

```
# 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
```

Verranno mostrati i nomi dei pacchetti da installare. (Ci potrebbero essere alcuni nomi non di pacchetto, come `"texmf"`.)

2.7 Suggerimenti per la gestione dei pacchetti

2.7.1 Come scegliere i pacchetti Debian

Si possono cercare i pacchetti che soddisfano i propri bisogni con `aptitude` in base alla descrizione del pacchetto o dall'elenco in `"Task"`.

Quando si incontrano più di due pacchetti simili e non si sa quale installare senza doverli installare per prova, si dovrebbe usare un po' di **buon senso**. Considero i punti seguenti un buon consiglio per i pacchetti da preferire.

- Essenziale: sì > no
 - Area: main > contrib > non-free
-

- Priorità: richiesto > importante > standard > opzionale > extra
- Task: pacchetti elencati in task come "Ambiente desktop"
- Packages selected by the dependency package (e.g., `gcc-10` by `gcc`)
- Popcon: maggior numero di voti e di installazioni
- Changelog: aggiornamenti regolari da parte del manutentore
- BTS: nessun bug RC (nessun bug critico, grave o serio)
- BTS: manutentore sensibile alle segnalazioni di bug
- BTS: maggior numero di bug risolti di recente
- BTS: minor numero di bug rimanenti non wishlist

Essendo Debian un progetto di volontari con modello di sviluppo distribuito, i suoi archivi contengono molti pacchetti con scopi e qualità molto differenti. Si deve scegliere con la propria testa che uso farne.

2.7.2 Pacchetti da fonti mescolate di archivi



Attenzione

Installare pacchetti da fonti mescolate di archivi non è supportato dalla distribuzione Debian ufficiale, tranne che per particolari combinazioni supportate ufficialmente, come `stable` con [security updates](#) e [bullseye-updates](#).

Ecco un esempio di operazioni da fare per includere, sporadicamente, versioni originali più recenti di specifici pacchetti presenti in `unstable` pur mantenendo `testing`.

1. Cambiare temporaneamente il file `/etc/apt/sources.list` ad una singola voce `unstable`.
2. Eseguire `aptitude update`.
3. Eseguire `aptitude install nome-pacchetto`.
4. Ripristinare il file `/etc/apt/sources.list` originale per `testing`.
5. Eseguire `aptitude update`.

Con questo approccio manuale non si deve creare il file `/etc/apt/preferences` né ci si deve preoccupare dei pin di `apt`. Ma è piuttosto laborioso.



Attenzione

Quando si usano fonti di archivi mescolate, si deve assicurare la compatibilità tra i pacchetti da soli, dato che Debian non la garantisce. Se esistono incompatibilità tra i pacchetti, si può danneggiare il sistema. Si deve essere in grado di giudicare queste esigenze tecniche. L'uso di fonti mescolate di archivi non correlati è una modalità di funzionamento del tutto facoltativa e il suo uso non è per nulla raccomandato.

Le regole generali per installare pacchetti da archivi diversi sono le seguenti.

- L'installazione di pacchetti non binari ("Architecture: all") è più sicura.
 - pacchetti di documentazione: nessun requisito particolare
 - pacchetti programma per interpreti: deve essere disponibile un interprete compatibile
-

- L'installazione di pacchetti binari (non "Architecture: all") di solito comporta molte difficoltà e non è sicura.
 - compatibilità di versione con librerie (inclusa "libc")
 - compatibilità di versione con programmi applicativi correlati
 - compatibilità con l'[ABI](#) del kernel
 - compatibilità con l'[ABI C++](#)
 - ...

Nota

Per rendere l'installazione di un pacchetto **più sicura**, possono essere forniti alcuni pacchetti binari applicativi commerciali non liberi con librerie completamente statiche. Si dovrebbe anche per essi controllare in ogni caso la compatibilità con [ABI](#), ecc.

Nota

Installare pacchetti binari da archivi non ufficialmente supportati è generalmente una cattiva idea, tranne che per evitare pacchetti danneggiati per un breve tempo. Questo è vero anche se si usano i pin di APT (vedere Sezione [2.7.3](#)). Si dovrebbe considerare chroot o tecniche simili (vedere Sezione [9.11](#)) per eseguire programmi da archivi differenti.

2.7.3 Modificare la versione candidata

**avvertimento**

L'uso dei pin di APT da parte di un utente inesperto è una fonte sicura di grossi problemi. Si dovrebbe evitare l'uso dei pin di APT se non strettamente necessario.

Senza il file `"/etc/apt/preferences"`, il sistema APT sceglie la versione più recente disponibile come **versione candidata** usando la stringa di versione. Questa è la situazione normale e l'uso raccomandato del sistema APT. Tutte le combinazioni di archivi supportate ufficialmente non necessitano del file `"/etc/apt/preferences"`, dato che gli archivi che non dovrebbero essere usati come fonti automatica per gli aggiornamenti sono marcati come **NotAutomatic** e trattati in modo appropriato.

Suggerimento

La regola di comparazione della stringa di versione può essere verificata con, ad esempio, `"dpkg --compare-versions ver1.1 gt ver1.1~1; echo $?"` (vedere `dpkg(1)`).

Quando si installano regolarmente pacchetti da fonti di archivi mescolate (vedere Sezione [2.7.2](#)), si possono automatizzare queste operazioni complicate creando il file `"/etc/apt/preferences"` con le voci appropriate e modificando la regola di selezione dei pacchetti per le **versioni candidate** come descritto in `apt_preferences(5)`. Questo è chiamato **usare i pin di APT**.

**Attenzione**

Quando si usano i pin di APT si deve assicurare la compatibilità dei pacchetti da soli dato che Debian non la garantisce. L'uso dei pin di APT è una modalità di funzionamento del tutto facoltativa e il suo uso non è per nulla consigliato.

**Attenzione**

Per le regole di `apt_preferences(5)` sono usati i file Release a livello di archivio (vedere Sezione [2.5.3](#)). I pin di APT perciò funzionano solo con il nome della "suite" per i [normali archivi Debian](#) e per gli [archivi di sicurezza Debian](#). (Ciò è diverso per gli archivi [Ubuntu](#).) Per esempio nel file `"/etc/apt/preferences"` si può usare `"Pin: release a=unstable"`, ma non `"Pin: release a=sid"`.



Attenzione

Quando si usano archivi non Debian con i pin di APT si dovrebbe controllare quale sia il loro scopo e la loro credibilità. Ubuntu e Debian, per esempio, non sono fatti per essere mescolati.

Nota

Anche se non si crea il file `"/etc/apt/preferences"` si possono fare operazioni piuttosto complesse (vedere Sezione 2.6.4 e Sezione 2.7.2) senza l'uso dei pin di APT.

Quella che segue è una spiegazione semplificata dell'uso dei pin di APT.

Il sistema APT sceglie, dalle fonti per i pacchetti disponibili definite nel file `"/etc/apt/sources.list"`, il pacchetto di **aggiornamento** con la priorità di pin più elevata come **versione candidata** del pacchetto. Se la priorità di pin del pacchetto è maggiore di 1000, la restrizione a versioni di **aggiornamento** non viene considerata per permettere la retrocessione ad una versione precedente (vedere Sezione 2.7.10).

Il valore della priorità di pin per ciascun pacchetto è definito dalle voci "Pin-Priority" nel file `"/etc/apt/preferences"` o viene usato il valore predefinito.

Priorità di pin	Effetti del pin di APT sul pacchetto
1001	installa il pacchetto anche se ciò costituisce il ritorno ad una versione precedente
990	usato come valore predefinito per l'archivio del rilascio prescelto
500	usato come valore predefinito per l'archivio normale
100	usato come valore predefinito per l'archivio NonAutomatico e MaConAggiornamentiAutomatici
100	usato per i pacchetti installati
1	usato come valore predefinito per l'archivio NonAutomatico
-1	non installa mai il pacchetto nemmeno se raccomandato

Tabella 2.18: Elenco di valori di priorità di pin che esemplificano la tecnica d'uso dei pin di APT.

L'archivio del **rilascio definito** può essere impostato in svariati modi.

- nel file di configurazione `"/etc/apt/apt.conf"` con la riga `"APT::Default-Release "stable";"`
- con l'opzione per la riga di comando, ad esempio, `"apt-get install -t testing qualche-pacchetto"`

Gli attributi di archivio **NonAutomatico** e **MaConAggiornamentiAutomatici** vengono impostati dal server dell'archivio che ha il file Release a livello di archivio (vedere Sezione 2.5.3) che contiene sia `"NotAutomatic: yes"` sia `"ButAutomaticUpgrades: yes"`. L'archivio **NonAutomatico** viene impostato dal server dell'archivio che ha il file Release a livello di archivio che contiene solamente `"NotAutomatic: yes"`.

La **situazione dei pin di APT** di *pacchetto* da diverse fonti di archivi viene mostrata con `"apt-cache policy pacchetto"`.

- Una riga che inizia con `"pacchetto pin:"` elenca la versione del pacchetto di **pin** se è definita un'associazione specifica per *pacchetto*, ad esempio `"Package pin: 0.190"`.
- Non esiste alcuna riga con `"Package pin:"` se non è definita alcuna associazione solamente con *pacchetto*.
- Il valore di priorità di pin associato solamente con *pacchetto* viene mostrato alla destra di tutte le stringhe di versione, ad esempio `"0.181 700"`.
- Se non è definita un'associazione specifica con *pacchetto* viene mostrato `"0"` alla destra di tutte le stringhe di versione.
- I valori di priorità di pin degli archivi (definiti come `"Package: *"` nel file `"/etc/apt/preferences"`) sono elencati alla sinistra di tutti i percorsi di archivio, ad esempio `"100 http://deb.debian.org/debian/bullseye-backports/main Packages"`.

2.7.4 Updates e Backports

Esistono gli archivi di [bullseye-updates](#) e [backports.debian.org](#) che forniscono pacchetti di aggiornamento per `stable` (`bullseye`).

Per usare questi archivi, si devono elencare tutti quelli richiesti nel file `"/etc/apt/sources.list"` nel modo seguente.

```
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
```

Non è necessario impostare esplicitamente un valore di priorità di pin nel file `"/etc/apt/preferences"`. Quando nuovi pacchetti diventano disponibili, la configurazione predefinita fornisce gli aggiornamenti più ragionevoli (vedere Sezione 2.5.3).

- Tutti i vecchi pacchetti installati vengono aggiornati con quelli più nuovi da `bullseye-updates`.
- Solo i vecchi pacchetti installati manualmente da `bullseye-backports` vengono aggiornati a quelli più nuovi da `bullseye-backports`.

Ogni volta che si desidera installare manualmente un pacchetto chiamato `"nome-pacchetto"` con le sue dipendenze dall'archivio `bullseye-backports`, usare questo comando che sposta il rilascio stabilito con l'opzione `"-t"`.

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

2.7.5 Bloccare i pacchetti da installare perché "Raccomandati"



avvertimento

L'uso dei pin di APT da parte di un utente inesperto è una fonte sicura di grossi problemi. Si dovrebbe evitare l'uso dei pin di APT se non strettamente necessario.

Se non si desidera installare pacchetti particolari richiamati automaticamente perché "Raccomandati", è necessario creare il file `"/etc/apt/preferences"` ed elencarvi esplicitamente all'inizio tutti quei pacchetti nel modo seguente.

```
Package: package-1
Pin: version *
Pin-Priority: -1
```

```
Package: package-2
Pin: version *
Pin-Priority: -1
```

2.7.6 Usare e aggiornare `testing` con alcuni pacchetti da `unstable`



avvertimento

L'uso dei pin di APT da parte di un utente inesperto è una fonte sicura di grossi problemi. Si dovrebbe evitare l'uso dei pin di APT se non strettamente necessario.

Quello che segue è un esempio di **uso dei pin di APT** per includere pacchetti specifici con versioni originali più recenti presenti in `unstable` aggiornandoli regolarmente pur mantenendo `testing`. Si devono elencare nel file `"/etc/apt/sources.list"` tutti gli archivi necessari nel modo seguente.

```
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
```

Impostare il file `"/etc/apt/preferences"` come segue.

```
Package: *
Pin: release a=unstable
Pin-Priority: 100
```

Quando, con questa configurazione, si desidera installare un pacchetto chiamato *"nome-pacchetto"* con le sue dipendenze dall'archivio `unstable`, usare il comando seguente che cambia il rilascio definito con l'opzione `"-t"` (la priorità di pin di `unstable` diventa 990).

```
$ sudo apt-get install -t unstable package-name
```

Con questa configurazione, la normale esecuzione di `"apt-get upgrade"` e `"apt-get dist-upgrade"` (o `"aptitude safe-upgrade"` e `"aptitude full-upgrade"`) aggiorna i pacchetti che erano stati installati dall'archivio `testing` usando l'attuale archivio `testing` e i pacchetti che erano stati installati dall'archivio `unstable` usando l'attuale archivio `unstable`.



Attenzione

Fare attenzione a non rimuovere la voce `"testing"` dal file `"/etc/apt/sources.list"`. Senza di essa il sistema APT aggiorna i pacchetti usando l'archivio `unstable` più recente.

Suggerimento

Io di solito modifico il file `"/etc/apt/sources.list"` per commentare le voci relative all'archivio `unstable` immediatamente dopo l'azione descritta sopra. Questo evita il rallentamento del processo di aggiornamento dovuto a troppe voci nel file `"/etc/apt/sources.list"` anche se impedisce l'aggiornamento, usando l'attuale archivio `unstable` dei pacchetti che erano stati installati dall'archivio `unstable`.

Suggerimento

Se viene usato `"Pin-Priority: 1"` invece di `"Pin-Priority: 100"` nel file `"/etc/apt/preferences"`, i pacchetti già installati che hanno 100 come valore della priorità di pin, non vengono aggiornati dall'archivio `unstable` nemmeno se viene rimossa la voce `"testing"` dal file `"/etc/apt/sources.list"`.

Se si desidera tenere traccia automaticamente di particolari pacchetti in `unstable` senza l'iniziale installazione con `"-t unstable"`, si deve creare il file `"/etc/apt/preferences"` ed elencarvi esplicitamente tutti i pacchetti voluti nel modo seguente.

```
Package: package-1
Pin: release a=unstable
Pin-Priority: 700
```

```
Package: package-2
Pin: release a=unstable
Pin-Priority: 700
```

Questo imposta il valore della priorità di pin per ciascun pacchetto specifico. Per esempio, per tenere traccia della versione più recente in `unstable` di questo manuale "Debian Reference" in inglese, si dovrebbero avere, nel file `"/etc/apt/preferences"` le voci seguenti.

```
Package: debian-reference-en
Pin: release a=unstable
Pin-Priority: 700
```

```
Package: debian-reference-common
Pin: release a=unstable
Pin-Priority: 700
```

Suggerimento

Questo modo di usare i pin di APT è valido anche quando si segue l'archivio `stable`. Per la mia esperienza, installare i pacchetti di documentazione dall'archivio `unstable` è sempre sicuro.

2.7.7 Usare e aggiornare `unstable` con alcuni pacchetti da `experimental`

Quello che segue è un esempio di **uso dei pin di APT** per includere pacchetti specifici con versioni originali più recenti presenti in `experimental` pur mantenendo `unstable`. Si devono elencare nel file `"/etc/apt/sources.list"` tutti gli archivi necessari nel modo seguente.

```
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
```

Il valore predefinito della priorità di pin per l'archivio `experimental` è sempre 1 ($\ll 100$) dato che è un archivio **Non automatico** (vedere Sezione 2.5.3). Per usare semplicemente l'archivio `experimental` non c'è alcun bisogno di impostare il valore di priorità di pin esplicitamente nel file `"/etc/apt/preferences"`, a meno che non si desideri aggiornare automaticamente da esso in futuro particolari pacchetti.

2.7.8 Scaricare ed aggiornare automaticamente i pacchetti



avvertimento

L'uso dei pin di APT da parte di un utente inesperto è una fonte sicura di grossi problemi. Si dovrebbe evitare l'uso dei pin di APT se non strettamente necessario.

Il pacchetto `apt` viene fornito con il proprio script di cron, `"/etc/cron.daily/apt"` per permettere di scaricare automaticamente i pacchetti. Questo script può essere migliorato, installando il pacchetto `unattended-upgrades`, per effettuare l'aggiornamento automatico dei pacchetti. Può essere personalizzato dai parametri in `"/etc/apt/apt.conf.d/02backup"` e `"/etc/apt/apt.conf.d/50unattended-upgrades"` come descritto in `"/usr/share/doc/unattended-upgrades/`

Il pacchetto `unattended-upgrades` è pensato principalmente per gli aggiornamenti di sicurezza del sistema `stable`. Se il rischio di danneggiare un sistema `stable` esistente con gli aggiornamenti automatici è minore di quello che il sistema venga danneggiato da un intruso, usando una falla nella sicurezza chiusa dall'aggiornamento di sicurezza, si dovrebbe considerare l'uso di questi aggiornamenti automatici con parametri simili ai seguenti.

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

Se si sta usando un sistema `unstable`, non è bene usare gli aggiornamenti automatici perché certamente prima o poi danneggeranno il sistema. Anche per `unstable` però, per risparmiare tempo nell'aggiornamento interattivo, si può volere scaricare i pacchetti in anticipo usando parametri di configurazione come i seguenti.

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

2.7.9 Limitare l'uso di banda di APT per gli scaricamenti

Se si desidera limitare l'uso della banda da parte di APT per gli scaricamenti, per esempio a 800Kib/sec (=100kiB/sec), si devono usare i parametri di configurazione di APT nel modo seguente.

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

2.7.10 Retrocessione di emergenza alla versione precedente



avvertimento

L'uso dei pin di APT da parte di un utente inesperto è una fonte sicura di grossi problemi. Si dovrebbe evitare l'uso dei pin di APT se non strettamente necessario.



Attenzione

La retrocessione ad una versione precedente non è ufficialmente supportata da Debian. Dovrebbe essere fatta solamente come parte di un processo di ripristino di emergenza. Nonostante questo è noto che funziona bene in caso di molti incidenti. Per i sistemi critici, si dovrebbe fare un backup di tutti i dati importanti sul sistema dopo l'operazione di ripristino e reinstallare un nuovo sistema da zero.

Si potrebbe essere fortunati e riuscire a retrocedere da un archivio più recente ad uno più vecchio per ripristinare un aggiornamento di sistema andato male, manipolando le **versioni candidate** (vedere Sezione 2.7.3). Questa è l'alternativa pigra al lavoro tedioso di dover usare molti comandi `dpkg -i pacchetto-danneggiato_vecchia_versione.deb` (vedere Sezione 2.6.4). Cercare le righe nel file `/etc/apt/sources.list` che rimandano a `unstable`, come la seguente.

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

Sostituirle con la seguente per puntare a `testing`.

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

Impostare il file `/etc/apt/preferences` come segue.

```
Package: *
Pin: release a=testing
Pin-Priority: 1010
```

Eseguire `apt-get update; apt-get dist-upgrade` per forzare la retrocessione dei pacchetti in tutto il sistema.

Rimuovere questo file speciale `/etc/apt/preferences` dopo la retrocessione di emergenza.

Suggerimento

È una buona idea rimuovere (non eliminare completamente!) il maggior numero di pacchetti per minimizzare i problemi di dipendenza. Potrebbe essere necessario rimuovere o installare manualmente alcuni pacchetti per retrocedere il sistema. Il kernel Linux, il bootloader, udev, PAM, APT e i pacchetti relativi alla rete ed i loro file di configurazione richiedono particolare attenzione.

2.7.11 Chi ha caricato il pacchetto?

Anche se i nomi dei manutentori elencati in `/var/lib/dpkg/available` e `/usr/share/doc/package_name/changes` forniscono alcune informazioni su "chi stia dietro la pacchettizzazione", l'identità di chi ha effettivamente caricato il pacchetto è un po' oscura. `who-uploads(1)` nel pacchetto `devscripts` identifica l'effettivo autore del caricamento dei pacchetti sorgenti Debian.

2.7.12 Il pacchetto equivs

Se si ha intenzione di compilare un programma dai sorgenti per rimpiazzare un pacchetto Debian, la cosa migliore è creare un vero pacchetto debianizzato locale (*.deb) e usare un archivio privato.

Se si sceglie di compilare un programma dai sorgenti e di installarlo invece in `"/usr/local"`, si può usare `equivs` come ultima spiaggia per soddisfare le dipendenze mancanti del pacchetto.

```
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 Fare il port di un pacchetto nel sistema stabile

Per aggiornamenti parziali del sistema `stable`, è una buona idea ricompilare un pacchetto, usando il pacchetto sorgente, all'interno del suo ambiente. Questo evita massicci aggiornamenti di pacchetti a causa delle dipendenze.

Aggiungere le voci seguenti al file `"/etc/apt/sources.list"` di un sistema `stable`.

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

Installare i pacchetti richiesti per la compilazione e scaricare il pacchetto sorgente, come nell'esempio seguente.

```
# 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*
```

Aggiornare alcuni pacchetti con catene di strumenti come `dpkg` e `debhelper` dai pacchetti backport se sono necessari per fare il backport.

Eseguire quanto segue.

```
$ dch -i
```

Incrementare la versione del pacchetto, ad esempio aggiungendo `"+bp1"` in `"debian/changelog"`

Compilare i pacchetti ed installarli nel sistema nel modo seguente.

```
$ debuild
$ cd ..
# debi foo*.changes
```

2.7.14 Server proxy per APT

Dato che fare il mirror di intere sottosezioni dell'archivio Debian spreca spazio su disco e banda, l'utilizzo di un server proxy locale per APT è una buona idea quando si amministrano molti sistemi in una LAN. APT può essere configurato per usare server proxy web generici (http), quale squid (vedere Sezione 6.5) come descritto in `apt.conf(5)` ed in `"/usr/share/doc/apt/examples/` La variabile d'ambiente `"$http_proxy"` può essere usata per scavalcare l'impostazione del server proxy nel file `"/etc/apt/apt.conf"`.

Ci sono strumenti proxy specifici per l'archivio Debian. Prima di usarli si dovrebbe controllare il BTS.

pacchetto	popcon	dimensione	descrizione
approx	V:0, I:0	6317	server proxy con cache per i file di archivio Debian (programma OCaml compilato)
apt-cacher	V:0, I:0	289	proxy con cache per file di pacchetto e sorgenti Debian (programma Perl)
apt-cacher-ng	V:5, I:5	1562	proxy con cache per la distribuzione di pacchetti software (programma in C++ compilato)

Tabella 2.19: Elenco degli strumenti proxy specifici per l'archivio Debian



Attenzione

Quando Debian riorganizza la struttura del suo archivio, questi strumenti proxy specializzati tendono ad aver bisogno di ritocchi al codice da parte del manutentore del pacchetto e possono non essere funzionanti per un po' di tempo. D'altra parte i server proxy web generici (http) sono più robusti e gestiscono questi cambiamenti più facilmente.

2.7.15 Ulteriori letture sulla gestione dei pacchetti

Si possono ottenere molte altre informazioni sulla gestione dei pacchetti dalla documentazione seguente.

- Documentazione principale sulla gestione dei pacchetti:
 - `aptitude(8)`, `dpkg(1)`, `tasksel(8)`, `apt(8)`, `apt-get(8)`, `apt-config(8)`, `apt-key(8)`, `sources.list(5)`, `apt.conf` e `apt_preferences(5)`;
 - `"/usr/share/doc/apt-doc/guide.html/index.html"` e `"/usr/share/doc/apt-doc/offline.html/index.html"` dal pacchetto `apt-doc` e
 - `"/usr/share/doc/aptitude/html/en/index.html"` dal pacchetto `aptitude-doc-en`.
- Documentazione ufficiale dettagliata sull'archivio Debian:
 - ["Manuale Debian Policy, capitolo 2 - L'archivio Debian"](#),
 - ["Debian Developer's Reference, capitolo 4 - Risorse per gli sviluppatori Debian, 4.6 L'archivio Debian"](#) e
 - ["FAQ Debian GNU/Linux, capitolo 6 - Gli archivi FTP Debian"](#).
- Tutorial per gli utenti Debian sulla creazione di un pacchetto Debian:
 - [«Guida per i manutentori Debian»](#).

Capitolo 3

Inizializzazione del sistema

È bene che l'amministratore di sistema conosca almeno a grandi linee come viene avviato e configurato il sistema Debian. Anche se i dettagli precisi sono nei file sorgenti dei pacchetti installati e nella loro documentazione, essi sono un po' troppo per la maggior parte degli utenti.

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)` describe il processo di avvio del sistema basato su `systemd` (Debian recenti).
- `boot(7)` describe il processo di avvio del sistema basato su UNIX System V Release 4 (Debian più vecchie).

3.1 Panoramica del processo di avvio

Il sistema del computer passa attraverso varie fasi del [processo di avvio](#), dall'accensione a quando offre all'utente il sistema operativo (SO) pienamente funzionante.

Per semplicità la spiegazione è limitata alla piattaforma PC tipica con l'installazione standard.

Il normale processo di avvio è come un razzo a quattro stadi. Ogni stadio del razzo passa il controllo del sistema allo stadio successivo.

- Sezione [3.1.1](#)
- Sezione [3.1.2](#)
- Sezione [3.1.3](#)
- Sezione [3.1.4](#)

Naturalmente questo può essere configurato in modo diverso. Per esempio, se è stato compilato un kernel personalizzato, si potrebbe saltare la fase con il mini-sistema Debian. Non dare per scontato che quanto detto valga per il proprio sistema fino a che non si abbia controllato direttamente.

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 Stadio 2: il bootloader

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.

pacchetto	popcon	dimensioni	initrd	bootloader	descrizione
grub-efi-amd64	I:214	158	Supportato	GRUB UEFI	This is smart enough to understand disk partitions and filesystems such as vfat, ext4, ... (UEFI)
grub-pc	V:26, I:757	533	Supportato	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	Supportato	GRUB 2	È l'immagine di ripristino avviabile di GRUB 2 (CD e floppy) (versione PC/BIOS)
lilo	V:0, I:2	697	Supportato	Lilo	Si basa sulla posizione dei settori dei dati sul disco fisso. (Vecchio)
syslinux	V:3, I:46	344	Supportato	Isolinux	Capisce il filesystem ISO 9660. Usato dal CD di avvio.
syslinux	V:3, I:46	344	Supportato	Syslinux	Capisce il filesystem MSDOS (FAT) . Usato dai dischetti floppy di avvio.
loadlin	V:0, I:1	90	Supportato	Loadlin	Il nuovo sistema viene avviato dal sistema FreeDOS/MSDOS.
mbr	V:0, I:6	50	Non supportato	MBR di Neil Turton	Software libero che sostituisce MBR MSDOS . Capisce solo le partizioni su disco.

Tabella 3.1: Elenco di bootloader



avvertimento

Non mettere mano ai bootloader senza aver creato supporti avviabili di ripristino (chiavette USB, CD o floppy) create da immagini nel pacchetto `grub-rescue-pc`. Questo permette di avviare il sistema anche senza un bootloader funzionante sul disco fisso.

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	valore
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

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

Suggerimento

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/`.

Vedere "info grub" e `grub-install(8)`.

3.1.3 Stadio 3: il mini-sistema Debian

Il mini-sistema Debian è il terzo stadio del processo di avvio che viene iniziato dal bootloader. Esegue il kernel del sistema con il suo filesystem root in memoria. Questo è uno stadio opzionale preparatorio del processo di avvio.

Nota

L'espressione "sistema Debian mini" è stata coniata per descrivere il terzo stadio del processo di avvio in questo documento. Normalmente ci si riferisce a questo sistema come sistema `initrd` o `initramfs`. Un sistema simile in memoria è usato dall'[installatore Debian](#).

Il programma `/init` viene eseguito come primo programma in questo file system root in memoria. È un programma che inizializza il kernel in spazio utente e passa il controllo allo stadio successivo. Questo mini-sistema Debian offre flessibilità al processo di avvio, come la possibilità di aggiungere moduli del kernel prima del processo di avvio principale o di montare il file system root come cifrato.

- Il programma `/init` è uno script di shell se `initramfs` è stato creato da `initramfs-tools`.
 - Si può interrompere questa parte del processo di avvio per ottenere una shell di root fornendo il parametro di avvio per il kernel `break=init` etc. Vedere lo script `/init` per ulteriori condizioni di interruzione. Questo ambiente shell è abbastanza sofisticato da fare una buona ispezione dell'hardware della macchina.
 - I comandi disponibili in questo mini-sistema Debian sono versioni minimali e vengono principalmente forniti da uno strumento GNU chiamato `busybox(1)`.
- Il programma `/init` è un programma binario di `systemd` se `initramfs` è stato creato da `dracut`.
 - I comandi disponibili in questo mini-sistema Debian sono un ambiente `systemd(1)` ridotto al minimo.



Attenzione

È necessario usare l'opzione `-n` per il comando `mount` quando si è nel filesystem root in sola lettura.

3.1.4 Stadio 4: il normale sistema Debian

Il sistema Debian normale è il quarto stadio del processo di avvio che viene iniziato dal mini-sistema Debian. Il kernel di sistema del mini-sistema Debian continua ad essere in esecuzione anche in questo ambiente. Il filesystem root viene cambiato da quello in memoria all'effettivo filesystem sul disco fisso.

Il programma `init` viene eseguito come primo programma con PID=1 per effettuare il processo principale di avvio di far partire molti programmi. Il percorso di file predefinito per il programma `init` è `«/sbin/init»`, ma può essere cambiato con un parametro di avvio del kernel come in `«init=/percorso/del/programma_init»`.

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

Suggerimento

Si può verificare quale è l'effettivo comando `init` nel proprio sistema con il comando `«ps --pid 1 -f»`.

pacchetto	popcon	dimensione	descrizione
systemd	V:826, I:928	16015	demone <code>init(8)</code> basato su eventi per concorrenza (alternativa a <code>sysvinit</code>)
systemd-sysv	V:818, I:925	139	le pagine di manuale e i collegamenti necessari affinché <code>systemd</code> sostituisca <code>sysvinit</code>
init-system-helpers	V:687, I:940	131	strumenti ausiliari per commutare tra <code>sysvinit</code> e <code>systemd</code>
initscripts	V:77, I:283	176	script per inizializzare ed arrestare il sistema
sysvinit-core	V:7, I:9	279	utilità <code>init(8)</code> in stile System-V
sysv-rc	V:154, I:296	81	meccanismo di cambiamento del runlevel in stile System-V
sysvinit-utils	V:459, I:999	79	utilità in stile System-V (<code>startpar(8)</code> , <code>bootlogd(8)</code> , ...)
lsb-base	V:890, I:999	49	funzionalità di script <code>init</code> Linux Standard Base 3.2
insserv	V:204, I:292	150	strumento per organizzare la sequenza di avvio usando dipendenze LSB negli script <code>init.d</code>
uswsusp	V:3, I:7	714	strumenti per usare la sospensione software in spazio utente fornita da Linux
kexec-tools	V:1, I:9	288	strumento <code>kexec</code> per riavvii <code>kexec(8)</code> (riavvio a caldo)
systemd-bootchart	V:0, I:1	128	analizzatore delle prestazioni del processo di avvio
bootchart2	V:0, I:0	94	analizzatore delle prestazioni del processo di avvio
pybootchartgui	V:0, I:0	177	analizzatore delle prestazioni del processo di avvio (visualizzazione)
mingetty	V:0, I:3	38	<code>getty(8)</code> solo console
mgetty	V:0, I:1	315	rimpiazzio di <code>getty(8)</code> per smart modem

Tabella 3.3: Elenco di utilità di avvio per il sistema Debian

Suggerimento

Vedere la [pagina del Wiki Debian sulla velocizzazione del processo di avvio](#) per i più recenti suggerimenti su come velocizzare il processo di avvio.

3.2 Init systemd

Questa sezione descrive come viene avviato il sistema dal programma `systemd(1)` con PID=1 (cioè processo `init`).

The `systemd` `init` process spawns processes in parallel based on the unit configuration files (see `systemd.unit(5)`) which are written in declarative style instead of SysV-like procedural style.

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 Sezione 4.7.4).

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

- `"/lib/systemd/system"`: file di configurazione predefiniti del sistema operativo
- `"/etc/systemd/system"`: file di configurazione dell'amministratore di sistema che scavalcano i file di configurazione predefiniti del sistema operativo
- `"/run/systemd/system"`: file di configurazione generati al momento dell'esecuzione che scavalcano i file di configurazione installati

Le loro inter-dipendenze sono specificate dalle direttive `"Wants="`, `"Requires="`, `"Before="`, `"After="`, ... (vedere `"MAPPING OF UNIT PROPERTIES TO THEIR INVERSES"` in `systemd.unit(5)`). Sono definiti anche i controlli delle risorse (vedere `systemd.resource-control(5)`).

Il suffisso dei file di configurazione delle unità codifica il loro tipo in questo modo:

- ***.service** descrive un processo controllato e supervisionato da `systemd`. Vedere `systemd.service(5)`.
- ***.device** descrive un device esposto in `sysfs(5)` come albero di device `udev(7)`. Vedere `systemd.device(5)`.
- ***.mount** descrive un punto di mount del file system controllato e supervisionato da `systemd`. Vedere `systemd.mount(5)`.
- ***.automount** descrive un punto di mount automatico del file system controllato e supervisionato da `systemd`. Vedere `systemd.automount(5)`.
- ***.swap** descrive un device o file di swap controllato e supervisionato da `systemd`. Vedere `systemd.swap(5)`.
- ***.path** descrive un percorso monitorato da `systemd` per l'attivazione basata su percorso. Vedere `systemd.path(5)`.
- ***.socket** descrive un socket controllato e supervisionato da `systemd` per l'attivazione basata su socket. Vedere `systemd.socket(5)`.
- ***.timer** descrive un timer controllato e supervisionato da `systemd` per l'attivazione basata su timer. Vedere `systemd.timer(5)`.
- ***.slice** gestisce risorse con `cgroups(7)`. Vedere `systemd.slice(5)`.
- ***.scope** viene creato programmaticamente usando le interfacce di bus di `systemd` per gestire un insieme di processi di sistema. Vedere `systemd.scope(5)`.
- ***.target** raggruppa altri file di configurazione di unità per creare punti di sincronizzazione durante l'avvio. Vedere `systemd.target(5)`.

All'avvio del sistema (cioè `init`) il processo `systemd` cerca di avviare `"/lib/systemd/system/default.target"` (normalmente un collegamento simbolico a `"graphical.target"`). Come prima cosa alcune speciali unità target (vedere `systemd.special(7)`) come `"local-fs.target"`, `"swap.target"` e `"cryptsetup.target"` sono richiamate per montare i file system. Poi altre unità target vengono anch'esse richiamate dalle dipendenze delle unità target. Per i dettagli leggere `bootup(7)`.

`systemd` offre funzionalità di compatibilità all'indietro. Gli script di avvio in stile SysV in `"/etc/init.d/rc[0123456S].d/[K]` sono comunque analizzati e `te limit(8)` viene tradotto in richieste di attivazione di unità `systemd`.



Attenzione

Run level emulati da 2 a 4 hanno tutti collegamenti simbolici al corrispondente `"multi-user.target"`.

3.2.1 Il nome host

Il kernel gestisce il **nome host** del sistema. L'unità `systemd` avviata da `systemd-hostnamed.service` imposta il nome host del sistema all'avvio al nome memorizzato in `"/etc/hostname"`. Questo file dovrebbe contenere **solamente** il nome host del sistema, non un nome di dominio pienamente qualificato.

Per visualizzare il nome host attuale eseguire `hostname(1)` senza alcun argomento.

3.2.2 Il filesystem

Le opzioni usate per montare i file system normali dei dischi e di rete sono impostate in `"/etc/fstab"`. Vedere `fstab(5)` e Sezione 9.6.7.

La configurazione dei file system cifrati è impostata in `"/etc/crypttab"`. Vedere `crypttab(5)`

Vedere `wvdial(1)` e `wvdial.conf(5)`.



avvertimento

Dopo aver montato tutti i filesystem, i file temporanei in `"/tmp"`, `"/var/lock"` e `"/var/run"` vengono ripuliti ad ogni avvio.

3.2.3 Inizializzazione delle interfacce di rete

Le interfacce di rete sono tipicamente inizializzate in `"networking.service"` per l'interfaccia `lo` e `"NetworkManager.service"` per le altre interfacce nei moderni sistemi desktop Debian che usano `systemd`.

Vedere Capitolo 5 per come configurarle.

3.3 I messaggi del kernel

I messaggi di errore del kernel visualizzati nella console possono essere configurati impostando il loro livello di soglia.

```
# dmesg -n3
```

valore del livello di errore	nome del livello di errore	significato
0	KERN_EMERG	il sistema è inutilizzabile
1	KERN_ALERT	bisogna agire immediatamente
2	KERN_CRIT	condizione critica
3	KERN_ERR	condizione di errore
4	KERN_WARNING	condizione di avvertimento
5	KERN_NOTICE	condizione normale ma significativa
6	KERN_INFO	messaggio informativo
7	KERN_DEBUG	messaggio a livello di debug

Tabella 3.4: Elenco dei livelli di errore del kernel

3.4 I messaggi di 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:

```
$ 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 Sezione 9.3.2.

Operazione	Esempio di 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>

Tabella 3.5: List of typical `journalctl` command snippets

3.5 System management

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

Negli esempi soprastanti "\$unit" può essere un singolo nome di unità (suffissi come `.service` e `.target` sono opzionali) o, in molti casi, la specifica di più unità (con glob in stile shell `"*"`, `"?"`, `"["`] usando `fnmatch(3)` con corrispondenze con i nomi primari di tutte le unità attualmente in memoria).

I comandi che cambiano lo stato del sistema negli esempi soprastanti sono tipicamente preceduti da `"sudo"` per ottenere i privilegi amministrativi necessari.

L'output di `"systemctl status $unit | $PID | $device"` usa il colore del puntino ("`●`") per riassumere lo stato dell'unità a prima vista.

- Un "`●`" bianco indica uno stato "inattivo" o "deattivato".
- Un "`●`" rosso indica uno stato di "fallimento" o "errore".
- Un "`●`" verde indica uno stato "attivo", "in ricaricamento" o "in attivazione".

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 Personalizzare systemd

3.7.1 Socket activation

Con l'installazione predefinita molti servizi di rete (vedere Capitolo 6) vengono avviati come processi demone dopo `network.target` al momento dell'avvio di sistema da `systemd`. `sshd` non fa eccezione. Come esempio di personalizzazione cambiamo questo comportamento nell'avvio on-demand di `sshd`.

Come prima cosa disabilitare l'unità di servizio installata dal sistema.

```
$ 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` per specificare un socket su cui restare in ascolto

Operazione	Esempio di comando
Elencare tutte le configurazioni delle unità target	<code>"systemctl list-units --type=target"</code>
Elencare tutte le configurazioni delle unità service	<code>"systemctl list-units --type=service"</code>
Elencare tutti i tipi di configurazioni delle unità	<code>"systemctl list-units --type=help"</code>
Elencare tutte le unità socket in memoria	<code>"systemctl list-sockets"</code>
Elencare tutte le unità timer in memoria	<code>"systemctl list-timers"</code>
Avviare "\$unit"	<code>"systemctl start \$unit"</code>
Fermare "\$unit"	<code>"systemctl stop \$unit"</code>
Ricaricare la configurazione specifica di un servizio	<code>"systemctl reload \$unit"</code>
Fermare e riavviare tutte le "\$unit"	<code>"systemctl restart \$unit"</code>
Avviare "\$unit" e fermare tutte le altre	<code>"systemctl isolate \$unit"</code>
Passare alla modalità "graphical" (sistema GUI)	<code>"systemctl isolate graphical"</code>
Passare alla modalità "multi-user" (sistema CLI)	<code>"systemctl isolate multi-user"</code>
Passare alla modalità "rescue" (sistema CLI a singolo utente)	<code>"systemctl isolate rescue"</code>
Inviare il segnale kill a "\$unit"	<code>"systemctl kill \$unit"</code>
Controllare se il servizio "\$unit" è attivo	<code>"systemctl is-active \$unit"</code>
Controllare se il servizio "\$unit" è fallito	<code>"systemctl is-failed \$unit"</code>
Controllare lo stato di "\$unit \$PID device"	<code>"systemctl status \$unit \$PID \$device"</code>
Mostrare le proprietà di "\$unit \$job"	<code>"systemctl show \$unit \$job"</code>
Ripristinare "\$unit" fallita	<code>"systemctl reset-failed \$unit"</code>
Elencare le dipendenze di tutti i servizi unità	<code>"systemctl list-dependencies --all"</code>
Elencare i file di unità installati sul sistema	<code>"systemctl list-unit-files"</code>
Abilitare "\$unit" (aggiungere collegamento simbolico)	<code>"systemctl enable \$unit"</code>
Disabilitare "\$unit" (rimuovere collegamento simbolico)	<code>"systemctl disable \$unit"</code>
Togliere maschera a "\$unit" (rimuovere collegamento simbolico a"/dev/null")	<code>"systemctl unmask \$unit"</code>
Mascherare "\$unit" (aggiungere collegamento simbolico a"/dev/null")	<code>"systemctl mask \$unit"</code>
Ottenere l'impostazione del target predefinito	<code>"systemctl get-default"</code>
Impostare default-target a "graphical" (sistema GUI)	<code>"systemctl set-default graphical"</code>
Impostare default-target a "multi-user" (sistema CLI)	<code>"systemctl set-default multi-user"</code>
Mostrare l'ambiente del lavoro	<code>"systemctl show-environment"</code>
Impostare la variabile "variable" dell'ambiente di lavoro al valore "value"	<code>"systemctl set-environment variable=value"</code>
Disimpostare la variabile "variable" dell'ambiente di lavoro	<code>"systemctl unset-environment variable"</code>
Ricaricare tutti i file di unità e i demoni	<code>"systemctl daemon-reload"</code>
Spegnere il sistema	<code>"systemctl poweroff"</code>
Spegnere e riavviare il sistema	<code>"systemctl reboot"</code>
Sospendere il sistema	<code>"systemctl suspend"</code>
Ibernare il sistema	<code>"systemctl hibernate"</code>

Tabella 3.6: List of typical systemctl command snippets

Operazione	Esempio di comando
Mostrare il tempo utilizzato per ogni passo di inizializzazione	"systemd-analyze time"
Elencare tutte le unità col tempo di inizializzazione	"systemd-analyze blame"
Carica e rileva gli errori nel file "\$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 cgrouops	"systemd-cgls"
Track boot process by the cgrouops	"ps xawf -eo pid,user,cgroup,args"
Track boot process by the cgrouops	Leggere sysfs in <code>"/sys/fs/cgroup/systemd/"</code>

Tabella 3.7: List of other monitoring command snippets under systemd

```
[Unit]
Description=SSH Socket for Per-Connection Servers

[Socket]
ListenStream=22
Accept=yes

[Install]
WantedBy=sockets.target
```

sshd@.service come file di servizio corrispondente di sshd.socket

```
[Unit]
Description=SSH Per-Connection Server

[Service]
ExecStart=-/usr/sbin/sshd -i
StandardInput=socket
```

Poi ricaricare.

```
$ sudo systemctl daemon-reload
```

3.8 Il 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 Sezione [1.2.12](#)), loads required kernel modules supporting it using the [modprobe\(8\)](#) program (see Sezione [3.8.1](#)), and creates corresponding device nodes.

Suggerimento

Se, per una qualche ragione `"/lib/modules/versione-kernel/modules.dep` non viene generato in modo appropriato dal `depmod(8)`, i moduli non possono essere caricati come dovuto dal sistema udev. Per risolvere il problema eseguire `"depmod -a"`.

Per le regole di montaggio in `"/etc/fstab"`, non è necessario che i nodi di device siano statici. Si possono usare gli [UUID](#) per montare i dispositivi, al posto dei nomi di device come `"/dev/sda"`. Vedere Sezione [9.6.3](#).

Dato che il sistema udev è in qualche modo in costante evoluzione, in questo documento sono fornite informazioni base, lasciando i dettagli ad altra documentazione.

3.8.1 L'inizializzazione dei moduli del kernel

Il programma `modprobe(8)` permette di configurare il kernel Linux in esecuzione da processi utente, aggiungendo e rimuovendo moduli del kernel. Il sistema udev (vedere Sezione 3.8) automatizza la sua invocazione per facilitare l'inizializzazione dei moduli del kernel.

Ci sono moduli non-hardware e speciali moduli con driver hardware, come quelli elencati in seguito, che devono essere precaricati elencandoli nel file `"/etc/modules"` (vedere `modules(5)`).

- moduli [TUN/TAP](#) che forniscono device virtuali di rete Point-to-Point (TUN) e device virtuali di rete Ethernet (TAP),
- moduli [netfilter](#) che forniscono funzionalità di firewall netfilter (`iptables(8)`, Sezione 5.6 e
- moduli driver [watchdog timer](#).

I file di configurazione per il programma `modprobe(8)` sono contenuti nella directory `"/etc/modprobes.d/"`, come spiegato in `modprobe.conf(5)`. (Se si desidera evitare l'autocaricamento di alcuni moduli del kernel, considerare la loro aggiunta nella lista nera nel file `"/etc/modprobes.d/blacklist"`.)

Il file `"/lib/modules/versione/modules.dep"` generato dal programma `depmod(8)` descrive le dipendenze dei moduli usate dal programma `modprobe(8)`.

Nota

Se si hanno problemi di caricamento dei moduli all'avvio o con `modprobe(8)`, `"depmod -a"` potrebbe risolverli rigenerando il file `"modules.dep"`.

Il programma `modinfo(8)` mostra informazioni su un modulo del kernel Linux.

Il programma `lsmod(8)` formatta in un bel modo i contenuti di `"/proc/modules"`, mostrando quali moduli del kernel siano attualmente caricati.

Suggerimento

Si può identificare l'esatto hardware sul proprio sistema. Vedere Sezione 9.5.3.

Si può configurare l'hardware all'avvio per attivare le funzionalità dell'hardware desiderate. Vedere Sezione 9.5.4.

Si può probabilmente aggiungere il supporto per il proprio speciale dispositivo ricompilando il kernel. Vedere Sezione 9.10.

Capitolo 4

Authentication and access controls

Quando una persona (o un programma) richiede l'accesso al sistema, l'autenticazione verifica che l'identità sia fidata.



avvertimento

Errori di configurazione di PAM possono lasciare l'utente fuori dal proprio sistema. Si deve avere un CD di ripristino a portata di mano o impostare una partizione di avvio alternativa. Per fare il ripristino, avviare il sistema con uno di essi e correggere le cose da lì.

4.1 Autenticazione Unix normale

L'autenticazione Unix normale è fornita dal modulo `pam_unix(8)` sotto [PAM \(Pluggable Authentication Modules\)](#). I suoi 3 importanti file di configurazione, con voci separate da ":", sono i seguenti.

file	permessi	utente	gruppo	descrizione
<code>/etc/passwd</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	informazioni sugli account utente (ripulite)
<code>/etc/shadow</code>	<code>-rw-r-----</code>	<code>root</code>	<code>shadow</code>	informazioni sicure sugli account utente
<code>/etc/group</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	informazioni sui gruppi

Tabella 4.1: I 3 importanti file di configurazione per `pam_unix(8)`

`/etc/passwd` contiene righe come le seguenti.

```
...
user1:x:1000:1000:User1 Name,,,:/home/user1:/bin/bash
user2:x:1001:1001:User2 Name,,,:/home/user2:/bin/bash
...
```

Come spiegato in `passwd(5)`, le voci separate da ":" in questo file hanno il significato seguente.

- Nome di login
- Voce di specificazione della password
- ID numerico dell'utente
- ID numerico del gruppo
- Nome dell'utente o campo di commento

- Directory home dell'utente
- Voce opzionale per l'interprete di comandi utente

La seconda voce del file `"/etc/passwd"` era usata per la password cifrata. Dopo l'introduzione di `"/etc/shadow"`, questa voce è usata per la specificazione della password.

contenuto	significato
(vuoto)	account senza password
x	la password cifrata è in <code>"/etc/shadow"</code>
!	nessun login per questo account
!	nessun login per questo account

Tabella 4.2: Il contenuto della seconda voce di `"/etc/passwd"`

`"/etc/shadow"` contiene righe come le seguenti.

```
...
user1:$1$Xop0FYH9$IfxYQwBe9b8tiyIkt2P4F/:13262:0:99999:7:::
user2:$1$vXGZLVbS$ElyErNf/agUDsm1DehJMS/:13261:0:99999:7:::
...
```

Come spiegato in `"shadow(5)`, le voci separate da `":` in questo file hanno il significato seguente.

- Nome di login
- Password cifrata (I caratteri `"1"` iniziali indicano l'uso della cifratura MD5. `"*"` indica nessun login.)
- Data dell'ultimo cambiamento di password, espressa come numero di giorni trascorsi dall'1 gennaio 1970
- Numero di giorni che l'utente deve aspettare prima di essere autorizzato a cambiare nuovamente la password
- Numero di giorni trascorsi i quali l'utente dovrà cambiare la propria password
- Numero di giorni di preavviso per avvisare l'utente che la password è in scadenza
- Numero di giorni dopo che una password è scaduta in cui la password viene sempre accettata
- Data di scadenza dell'account, espressa come numero di giorni trascorsi dall'1 gennaio 1970.
- ...

`"/etc/group"` contiene righe come le seguenti.

```
group1:x:20:user1,user2
```

Come spiegato in `"group(5)`, le voci separate da `":` in questo file hanno il significato seguente.

- Nome del gruppo
- Password cifrata (non realmente usato)
- ID numerico del gruppo
- Elenco di nomi utente separati da `","`

Nota

`"/etc/gshadow"` fornisce funzioni simili a `"/etc/shadow"` per `"/etc/group"`, ma non è realmente usato.

Nota

Può essere dinamicamente aggiunta la reale appartenenza di un utente ad un gruppo se viene aggiunta la riga "auth optional pam_group.so al file `/etc/pam.d/common-auth`" e se viene impostata in `/etc/security/group.conf`". Vedere `pam_group(8)`.

Nota

Il pacchetto `base-passwd` contiene una lista autorevole di utenti e gruppi: `/usr/share/doc/base-passwd/users-and-groups.html`.

4.2 Gestire le informazioni su account e password

Quelli che seguono sono alcuni comandi degni di nota per gestire le informazioni sugli account

comando	funzione
<code>getent passwd nome_utente</code>	sfoglia le informazioni sull'account di " <code>nome_utente</code> "
<code>getent shadow nome_utente</code>	sfoglia le informazioni shadow sull'account di " <code>nome_utente</code> "
<code>getent group nome_gruppo</code>	sfoglia le informazioni sul gruppo " <code>nome_gruppo</code> "
<code>passwd</code>	gestisce la password per l'account
<code>passwd -e</code>	imposta una password usa e getta per l'attivazione dell'account
<code>chage</code>	gestisce le informazioni sulla scadenza della password

Tabella 4.3: Elenco di comandi per gestire informazioni su account

Può essere necessario avere i privilegi di root per far funzionare alcune di queste funzioni. Vedere `crypt(3)` per la cifratura di password e dati.

Nota

Nei sistemi configurati con PAM e NSS, come la macchina Debian [salsa](#) Debian, il contenuto dei file locali `/etc/passwd`, `/etc/group` ed `/etc/shadow` può non essere attivamente usato dal sistema. I comandi sopra descritti sono validi anche in un ambiente di questo tipo.

4.3 Password buone

Quando si crea un account, durante l'installazione del sistema o con il comando `passwd(1)`, si dovrebbe scegliere una [buona password](#) che consiste, secondo `passwd(1)`, di un numero di caratteri da almeno 6 a 8, incluso uno o più caratteri da ciascuno dei seguenti insiemei.

- Lettere dell'alfabeto minuscole
- Cifre da 0 a 9
- Segni di punteggiatura

**avvertimento**

Non scegliere parole indovinabili come password. Il nome dell'account, il codice fiscale, il numero di telefono, l'indirizzo, il giorno del compleanno, il nome di membri della propria famiglia o di animali, parole del dizionario, semplici sequenze di caratteri come "12345" o "qwerty", ...sono tutte scelte pessime come password.

4.4 Creare password cifrate

Ci sono strumenti indipendenti per [generare password cifrate con salt](#).

pacchetto	popcon	dimensione	comando	funzione
whois	V:32, I:371	364	mkpasswd	frontend ricco di funzionalità per la libreria crypt(3)
openssl	V:814, I:994	1465	openssl passwd	calcola hash di password (OpenSSL). passwd(1ssl)

Tabella 4.4: Elenco di strumenti per generare password

4.5 PAM e NSS

I moderni sistemi **nix* come il sistema Debian, forniscono all'amministratore di sistema i meccanismi [PAM \(Pluggable Authentication Modules\)](#) e [NSS \(Name Service Switch\)](#) per configurare il sistema. Il ruolo di questi meccanismi può essere riassunto nel modo seguente.

- PAM offre un meccanismo di autenticazione flessibile usato dal software applicativo e comporta pertanto scambio di dati sulle password
- NSS offre un meccanismo di servizio dei nomi flessibile che è usato di frequente dalla [libreria standard C](#) per ottenere i nomi di utenti e gruppi per programmi come `ls(1)` e `id(1)`.

Questi sistemi PAM e NSS devono essere configurati in modo coerente.

I pacchetti degni di nota relativi ai sistemi PAM e NSS sono i seguenti.

pacchetto	popcon	dimensione	descrizione
libpam-modules	V:813, I:999	1048	Pluggable Authentication Modules, moduli di autenticazione inseribili (servizio base)
libpam-ldap	I:12	249	Pluggable Authentication Module, modulo di autenticazione inseribile, che permette interfacce LDAP
libpam-cracklib	I:15	117	Pluggable Authentication Module, modulo di autenticazione inseribile, che permette il supporto di cracklib
libpam-systemd	V:484, I:869	574	Pluggable Authentication Module, modulo di autenticazione inseribile, per registrare sessioni utente per <code>logind</code>
libpam-doc	I:1	1046	Pluggable Authentication Modules, moduli di autenticazione inseribili (documentazione in html e testo semplice)
libc6	V:946, I:999	12772	Libreria GNU C: librerie condivise che forniscono anche il servizio "Name Service Switch"
glibc-doc	I:11	3161	Libreria GNU C: pagine man
glibc-doc-reference	I:4	12740	Libreria GNU C: manuale di riferimento nei formati info, pdf e html (non-free)
libnss-mdns	I:508	150	Modulo NSS per la risoluzione dei nomi DNS Multicast
libnss-ldap	I:11	265	Modulo NSS per l'uso di LDAP come servizio per i nomi
libnss-ldapd	I:14	153	Modulo NSS per l'uso di LDAP come servizio per i nomi (nuovo fork di <code>libnss-ldap</code>)

Tabella 4.5: Elenco dei pacchetti degni di nota per i sistemi PAM e NSS

- La guida per l'amministratore di sistema di Linux-PAM, "The Linux-PAM System Administrators' Guide", in `libpam-doc` è essenziale per imparare la configurazione di PAM.

- La sezione "System Databases and Name Service Switch" in [glibc-doc-reference](#) è essenziale per imparare la configurazione di NSS.

Nota

Si può ottenere un elenco più esteso ed aggiornato usando il comando "aptitude search 'libpam-|libnss-'" . L'acronimo NSS può anche significare "Network Security Service" che è una cosa diversa da "Name Service Switch".

Nota

PAM è il metodo più basilare per inizializzare le variabili d'ambiente per ciascun programma con valori predefiniti a livello di sistema.

Con [systemd](#), il pacchetto `libpam-systemd` viene installato per gestire il login degli utenti registrando le sessioni utente nella gerarchia dei gruppi di controllo di `systemd` per [logind](#). Vedere `systemd-logind(8)`, `logind.conf(5)` e `pam-systemd(8)`.

4.5.1 File di configurazione letti da PAM e NSS

Quelli che seguono sono alcuni file degni di nota letti da PAM e NSS.

file di configurazione	funzione
<code>/etc/pam.d/nome_programma</code>	imposta la configurazione PAM per il programma " <code>nome_programma</code> "; vedere <code>pam(7)</code> e <code>pam.d(5)</code>
<code>/etc/nsswitch.conf</code>	imposta la configurazione NSS con la voce per ciascun servizio. Vedere <code>nsswitch.conf(5)</code>
<code>/etc/nologin</code>	limita il login utente con il modulo <code>pam_nologin(8)</code>
<code>/etc/securetty</code>	limita la tty per l'accesso root con il modulo <code>pam_securetty(8)</code>
<code>/etc/security/access.conf</code>	imposta il limite di accesso con il modulo <code>pam_access(8)</code>
<code>/etc/security/group.conf</code>	imposta restrizioni basate su gruppi con il modulo <code>pam_group(8)</code>
<code>/etc/security/pam_env.conf</code>	imposta le variabili d'ambiente con il modulo <code>pam_env(8)</code>
<code>/etc/environment</code>	imposta variabili d'ambiente aggiuntive con il modulo <code>pam_env(8)</code> con l'argomento " <code>readenv=1</code> "
<code>/etc/default/locale</code>	imposta la localizzazione con il modulo <code>pam_env(8)</code> con l'argomento " <code>readenv=1 envfile=/etc/default/locale</code> " (Debian)
<code>/etc/security/limits.conf</code>	imposta limitazioni alle risorse (<code>ulimit</code> , <code>core</code> , ...) con il modulo <code>pam_limits(8)</code>
<code>/etc/security/time.conf</code>	imposta limitazioni temporali con il modulo <code>pam_time(8)</code>
<code>/etc/systemd/logind.conf</code>	imposta la configurazione del gestore di login di <code>systemd</code> (vedere <code>logind.conf(5)</code> e <code>systemd-logind.service(8)</code>)

Tabella 4.6: Elenco di file di configurazione letti da PAM e NSS

Le restrizioni sulla scelta delle password sono implementate dai moduli PAM `pam_unix(8)` e `pam_cracklib(8)`. Possono essere configurati tramite i loro argomenti.

Suggerimento

I moduli PAM usano il suffisso ".so" nei loro nomi file.

4.5.2 La moderna gestione centralizzata del sistema

La moderna gestione centralizzata del sistema può essere messa in atto usando il server del [Protocollo LDAP \(Lightweight Directory Access Protocol\)](#) per amministrare molti sistemi *nix e non *nix in rete. L'implementazione open source del protocollo LDAP è il [software OpenLDAP](#).

Il server LDAP fornisce le informazioni sugli account attraverso l'uso di PAM e NSS con i pacchetti per il sistema Debian `libpam-ldap` e `libnss-ldap`. Per abilitare ciò sono necessarie diverse azioni. (Non ho mai usato questa configurazione e le informazioni che seguono sono di seconda mano. Tenerlo a mente quando si legge quanto segue.)

- Si configura un server LDAP centralizzato eseguendo un programma come il demone LDAP autonomo `slapd(8)`.
- Si modificano i file di configurazione di PAM nella directory `"/etc/pam.d/"` per usare `"pam_ldap.so"` invece del predefinito `"pam_unix.so"`.
 - Debian usa `"/etc/pam_ldap.conf"` come file di configurazione per `libpam-ldap` e `"/etc/pam_ldap.secret"` come file per archiviare la password di root.
- Si modifica la configurazione di NSS nel file `"/etc/nsswitch.conf"` per usare `"ldap"` invece della scelta predefinita (`"compat"` o `"file"`).
 - Debian usa `"/etc/libnss-ldap.conf"` come file di configurazione per `libnss-ldap`.
- Per la sicurezza delle password è necessario far sì che `libpam-ldap` usi una connessione [SSL \(o TLS\)](#).
- Per assicurare l'integrità dei dati, si può far sì che `libnss-ldap` usi una connessione [SSL \(o TLS\)](#) a prezzo di un maggiore carico sulla rete LDAP.
- Si dovrebbe eseguire `nscd(8)` localmente per mettere nella cache ogni risultato di ricerche LDAP in modo da ridurre il traffico di rete LDAP.

Vedere la documentazione in `pam_ldap.conf(5)` e `"/usr/share/doc/libpam-doc/html/"` fornita dal pacchetto `libpam-doc` e in `"info libc 'Name Service Switch'"` fornita dal pacchetto `libc-doc`.

In modo simile si possono impostare sistemi centralizzati alternativi con altri metodi.

- Integrazione di utenti e gruppi con il sistema Windows.
 - Accesso ai servizi [di dominio Windows](#) con i pacchetti `winbind` e `libpam_winbind`.
 - Vedere `winbindd(8)` e [Integrare reti MS Windows con Samba](#).
- Integrazione di utenti e gruppi con sistemi simil-Unix datati
 - Accesso a [NIS \(chiamato in origine YP\)](#) o [NIS+](#) con il pacchetto `nis`.
 - Vedere il [Linux NIS\(YP\)/NYS/NIS+ HOWTO](#).

4.5.3 "Perché GNU su non supporta il gruppo wheel"

Questa è la famosa sezione scritta da Richard M. Stallman, alla fine della vecchia pagina `"info su"`. Non c'è da preoccuparsi: l'attuale comando `SU` in Debian usa PAM, perciò questo può limitare la possibilità di usare `SU` verso il gruppo `root` abilitando la riga con `"pam_wheel.so"` in `"/etc/pam.d/su"`.

4.5.4 Regole più stringenti per le password

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 Sicurezza dell'autenticazione

Nota

Le informazioni fornite in questo documento **potrebbero non essere sufficienti** per le proprie necessità di sicurezza ma dovrebbero essere un **buon punto di partenza**.

4.6.1 Password sicure in Internet

Molti servizi per livello di trasporto popolari comunicano i loro messaggi, compresa l'autenticazione con password, in puro testo. È una pessima idea trasmettere password in puro testo attraverso l'Internet selvaggia dove possono essere intercettate. Si possono eseguire questi servizi attraverso "TLS" (Transport Layer Security, sicurezza del livello di trasporto), o il suo predecessore "SSL" (Secure Sockets Layer, livello per socket sicuri), per rendere sicura tramite cifratura tutta la comunicazione, compresa la password.

nome del servizio non sicuro	porta	nome del servizio sicuro	porta
www (http)	80	https	443
smtp (posta)	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

Tabella 4.7: Elenco di servizi e porte sicuri e non sicuri

La cifratura ha un costo in termini di tempo CPU. Come alternativa leggera per la CPU, si può mantenere la comunicazione in testo semplice, rendendo allo stesso tempo sicura la sola password con un protocollo di autenticazione sicura come "APOP" (Authenticated Post Office Protocol) per POP e "CRAMD-MD5" (Challenge-Response Authentication Mechanism MD5) per SMTP e IMAP. (Per inviare messaggi di posta elettronica via Internet dal proprio programma di posta al proprio server di posta è diventato popolare recentemente l'uso per SMTP della porta 587 invece della porta tradizionale 25, per evitare il blocco da parte del fornitore del servizio Internet della porta 25 autenticandosi allo stesso tempo con CRAM-MD5.)

4.6.2 Secure Shell, shell sicura

Il programma [Secure Shell \(SSH\)](#) fornisce comunicazioni sicure cifrate tra due host non fidati attraverso una rete non sicura, grazie ad un'autenticazione sicura. Consiste del client [OpenSSH](#), ssh(1) e del demone [OpenSSH](#), sshd(8). SSH può essere usato per fare da tunnel sicuro attraverso Internet per le comunicazioni con protocollo non sicuro come POP ed X, con la funzionalità di inoltrare delle porte.

Il client cerca di autenticarsi usando un'autenticazione basata sull'host, su una chiave pubblica, challenge-response o con password. L'uso di un'autenticazione con chiave pubblica permette il login remoto senza password. Vedere Sezione [6.3](#).

4.6.3 Misure aggiuntive di sicurezza per Internet

Anche quando si eseguono servizi sicuri, come server [SSH \(Secure shell\)](#) e [PPTP \(Point-to-Point Tunneling Protocol\)](#), esiste sempre la possibilità di un'intrusione da Internet con attacchi basati sull'indovinare la password usando metodi con forza bruta, ecc. L'uso di una politica di firewall (vedere Sezione [5.6](#)) insieme agli strumenti di sicurezza elencati in seguito, può migliorare la sicurezza generale.

pacchetto	popcon	dimensione	descrizione
knockd	V:0, I:3	102	demone, knockd(1) , e client, knock(1) piccoli per bussare alle porte
fail2ban	V:116, I:129	2092	strumento per interdire IP che causano errori di autenticazione multipli
libpam-shield	V:0, I:0	115	blocca all'esterno gli attacchi remoti che cercano di indovinare le password

Tabella 4.8: Elenco di strumenti per fornire misure aggiuntive di sicurezza

4.6.4 Rendere sicura la password di root

Per impedire che qualcuno possa accedere alla propria macchina con privilegi di root, è necessario compiere le azioni seguenti.

- Impedire l'accesso fisico al disco fisso
- Lock UEFI/BIOS and prevent booting from the removable media
- Impostare una password per la sessione interattiva di GRUB
- Bloccare il menu di GRUB impedendo i cambiamenti

Avendo accesso fisico al disco fisso, reimpostare la password di root è relativamente semplice seguendo i passi seguenti.

1. Move the hard disk to a PC with CD bootable UEFI/BIOS.
2. Avviare il sistema con un supporto di ripristino (disco di avvio di Debian, CD Knoppix, CD GRUB, ...).
3. Montare la partizione root con accesso in lettura e scrittura
4. Modificare il file `"/etc/passwd"` nella partizione root e rendere la seconda voce per l'account di `root` vuota.

Se si ha l'accesso in modifica alle voci di menu di GRUB (vedere Sezione [3.1.2](#)) per `grub-rescue-pc` all'avvio, è ancora più semplice, seguendo i passi seguenti.

1. Avviare il sistema con i parametri del kernel modificati in qualcosa del tipo `"root=/dev/hda6 rw init=/bin/sh"`.
2. Modificare il file `"/etc/passwd"` e rendere la seconda voce per l'account di `root` vuota.
3. Riavviare il sistema.

Si può ora accedere alla shell di root del sistema senza password.

Nota

Una volta ottenuto l'accesso alla shell di root, si ha l'accesso a qualsiasi cosa sul sistema e si può reimpostare qualsiasi password. Inoltre, si possono compromettere le password per tutti gli account utente usando strumenti di violazione delle password con attacchi a forza bruta, come quelli nei pacchetti `john` e `crack` (vedere Sezione [9.5.11](#)). Queste password violate possono portare alla compromissione di altri sistemi.

L'unica soluzione software ragionevole per evitare tutte queste preoccupazioni è l'uso di una partizione root (o partizione `"/etc"`) cifrata, usando [dm-crypt](#) e `initramfs` (vedere Sezione [9.9](#)). Tuttavia è sempre necessaria la password per avviare il sistema.

4.7 Altri controlli sugli accessi

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

Nota

Vedere Sezione [9.4.15](#) per limitare la funzionalità [SAK \(Secure Attention Key\)](#) del kernel.

4.7.1 sudo

`sudo(8)` è un programma progettato per permettere ad un amministratore di sistema di dare privilegi di root limitati ad utenti, e di registrare l'attività come root. `SUDO` richiede solo la password di un utente regolare. Installare il pacchetto `sudo` e attivarlo impostando le opzioni in `/etc/sudoers`. Vedere esempi di configurazione in `/usr/share/doc/sudo/examples/sudoers` e Sezione [1.1.12](#).

Il mio uso di `sudo` per un sistema con un singolo utente (vedere Sezione [1.1.12](#)) è mirato a proteggere me stesso dalla mia stupidità. Personalmente condidero l'uso di `sudo` come un'alternativa migliore all'uso costante del sistema dall'account root. Per esempio, il comando seguente cambia il proprietario di `"un_certo_file"` in `"mio_nome"`.

```
$ sudo chown my_name some_file
```

Naturalmente se si conosce la password di root (come accade per ogni utente Debian che ha installato il proprio sistema), qualsiasi comando può essere eseguito come utente root da qualsiasi account utente usando `"su -c"`.

4.7.2 PolicyKit

[PolicyKit](#) è un componente del sistema operativo per controllare privilegi a livello di sistema in sistemi operativi simil-Unix.

Le applicazioni GUI più recenti non sono pensate per essere eseguite come processi privilegiati. Per effettuare operazioni amministrative comunicano con processi privilegiati attraverso [PolicyKit](#).

[PolicyKit](#) limita tali operazioni agli account utente che appartengono al gruppo `sudo`, in sistemi Debian.

Vedere `polkit(8)`.

4.7.3 Limitare l'accesso ad alcuni servizi server

Per la sicurezza del sistema è una buona idea disabilitare il maggior numero di programmi server possibile. Questo aspetto diventa critico per i server di rete. Avere server inutilizzati, attivati direttamente come [demoni](#) o attraverso un programma [super-server](#), è considerato un rischio per la sicurezza.

Molti programmi, come `sshd(8)`, usano un controllo degli accessi basato su [PAM](#). Ci sono molti modi per limitare gli accessi ad un qualche servizio server.

- file di configurazione: `/etc/default/nome_programma`
- Systemd service unit configuration for [daemon](#)
- [PAM \(Pluggable Authentication Modules\)](#)
- `/etc/inetd.conf` per il [super-server](#)
- `/etc/hosts.deny` e `/etc/hosts.allow` per il [wrapper TCP](#), `tcpd(8)`
- `/etc/rpc.conf` per [Sun RPC](#)
- `/etc/at.allow` e `/etc/at.deny` per `atd(8)`

- `"/etc/cron.allow"` e `"/etc/cron.deny"` per `crontab(1)`
- il [firewall di rete](#) della infrastruttura [netfilter](#)

See Sezione 3.5, Sezione 4.5.1, and Sezione 5.6.

Suggerimento

I servizi [Sun RPC](#) devono essere attivi per i programmi [NFS](#) ed altri programmi basati su RPC.

Suggerimento

Se si hanno problemi con l'accesso remoto in sistemi Debian recenti, commentare la configurazione responsabile come "ALL: PARANOID" in `"/etc/hosts.deny"`, se esiste. (Essere però consapevoli dei rischi per la sicurezza che questo tipo di azione comporta.)

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

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

Capitolo 5

Impostazione della rete

Suggerimento

Per una guida moderna sull'uso della rete specifica per Debian, leggere [The Debian Administrator's Handbook — Configuring the Network](#).

Suggerimento

In [systemd](#), si può usare [networkd](#) per gestire il le reti. Vedere `systemd-networkd(8)`.

5.1 L'infrastruttura base di rete

In questa sezione viene descritta in breve l'infrastruttura base di rete in un sistema Debian moderno.

5.1.1 Risoluzione dei nomi di host

La risoluzione del nome host è attualmente supportata dal meccanismo [NSS \(Name Service Switch\)](#). Il flusso di eventi nella risoluzione è il seguente.

1. Il file `"/etc/nsswitch.conf"` con blocchi tipo `"hosts: files dns"` detta l'ordine di risoluzione dei nomi di host. (Questo rimpiazza la vecchia funzionalità del blocco `"order in "/etc/host.conf"`.)
2. Il metodo `files` è invocato per primo. Se il nome host viene trovato nel file `"/etc/hosts"`, vengono restituiti tutti gli indirizzi validi per esso e il programma termina. (Il file `"/etc/host.conf"` contiene `"multi on"`.)
3. Viene invocato il metodo `dns`. Se il nome host viene trovato dalla interrogazione al [DNS \(Internet Domain Name System, sistema dei nomi di dominio Internet\)](#), identificato dal file `"/etc/resolv.conf"`, vengono restituiti tutti gli indirizzi validi per esso e il programma termina.

Per esempio, `"/etc/hosts"` contiene righe come le seguenti.

```
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
```

pacchetto	popcon	dimensioni	tipo	descrizione
network-manager	V:360, I:426	15210	config::NM	NetworkManager (demone): gestisce la rete automaticamente
network-manager-gnome	V:124, I:357	5351	config::NM	NetworkManager (frontend GNOME)
ifupdown	V:612, I:989	217	config::ifupdown	strumento standard per attivare e disattivare la rete (specifico di Debian)
isc-dhcp-client	V:222, I:980	686	config::low-level	client DHCP
pppoeconf	V:0, I:8	192	config::helper	strumento di aiuto di configurazione per la connessione PPPoE
wpasupplicant	V:335, I:491	3439	” ”	supporto client per WPA e WPA2 (IEEE 802.11i)
wpaui	V:0, I:2	780	” ”	client GUI Qt per WPA supplicant
wireless-tools	V:180, I:239	297	” ”	strumenti per manipolare le estensioni wireless per Linux
iw	V:320, I:467	293	” ”	tool for configuring Linux wireless devices
iproute2	V:695, I:937	2867	config::iproute2	iproute2 , IPv6 e altra configurazione di rete avanzata: ip(8) , tc(8) , ecc.
iptables	V:314, I:990	2521	config::Netfilter	strumenti amministrativi per filtraggio di pacchetti di rete e NAT (Netfilter)
iputils-ping	V:224, I:997	113	test	testa la raggiungibilità di rete di un host remoto con nome host o indirizzo IP (iproute2)
iputils-arping	V:7, I:98	55	test	testa la raggiungibilità di rete di un host remoto specificato con il suo indirizzo ARP
iputils-tracert	V:4, I:53	72	test	traccia il percorso di rete verso un host remoto
ethtool	V:103, I:266	597	test	mostra e modifica le impostazioni dei device Ethernet
mtr-tiny	V:6, I:55	161	test::low-level	traccia il percorso di rete verso un host remoto (curses)
mtr	V:4, I:46	214	” ”	traccia il percorso di rete verso un host remoto (curses e GTK)
gnome-nettool	V:1, I:38	2105	” ”	strumenti per operazioni comuni sulle informazioni di rete (GNOME)
nmap	V:27, I:246	4509	” ”	strumento per mappatura della rete / esplorazione delle porte (Nmap , console)
zenmap	V:1, I:8	2939	” ”	strumento per mappatura della rete /esplorazione delle porte (GTK)
tcpdump	V:19, I:208	1329	” ”	analizzatore del traffico di rete (Tcpdump , console)
wireshark	I:54	65	” ”	analizzatore del traffico di rete (Wireshark , GTK)
tshark	V:3, I:33	408	” ”	analizzatore del traffico di rete (console)
tcptrace	V:0, I:3	401	” ”	produce un riassunto delle connessioni dall'output di tcpdump
snort	V:0, I:1	2206	” ”	sistema flessibile di rilevazione delle intrusioni di rete (Snort)
ntopng	V:1, I:2	969	” ”	mostra l'uso della rete nel browser web
dnsutils	V:59, I:490	261	” ”	client di rete forniti con BIND : nslookup(8) , nsupdate(8) , dig
dlint	V:0, I:6	53	” ”	controlla l'informazione DNS di zona usando interrogazioni del server di nomi
dnstracer	V:0, I:1	61	” ”	traccia una catena di server DNS fino alla sorgente

Tabella 5.1: Elenco degli strumenti di configurazione della rete

Ogni riga inizia con un [indirizzo IP](#) e prosegue con l'[hostname](#) associato.

L'indirizzo IP `127.0.1.1` nella seconda riga dell'esempio può non essere presente in altri sistemi simil-Unix. L'[Installatore Debian](#) crea questa voce per i sistemi senza un indirizzo IP permanente come soluzione per alcuni software (es., GNOME) come documentato nel [bug#719621](#).

Il `nome_host` corrisponde al nome host definito in `"/etc/hostname"`.

Per un sistema con un indirizzo IP permanente, si dovrebbe usare qui tale indirizzo invece di `127.0.1.1`.

Per un sistema con un indirizzo IP permanente e un [FQDN \(Fully Qualified Domain Name, nome di dominio pienamente qualificato\)](#) fornito dal [DNS \(Domain Name System\)](#), dovrebbe essere usato qui il canonico `nome_host.nome_dominio`, invece del semplice `nome_host`.

Il file `"/etc/resolv.conf"` è un file statico se non è installato il pacchetto `resolvconf`. Se invece quest'ultimo è installato il file è un collegamento simbolico. In ogni caso contiene le informazioni che inizializzano le routine del risolutore. Se il DNS si trova all'IP=`"192.168.11.1"`, il file conterrà la riga seguente.

```
nameserver 192.168.11.1
```

Il pacchetto `resolvconf` rende questo file `"/etc/resolv.conf"` un collegamento simbolico e gestisce il suo contenuto con script eseguiti in automatico.

Per le postazioni PC nei tipici ambienti LAN ad hoc, il nome host può essere risolto usando il Multicast DNS (mDNS, [Zeroconf](#)) in aggiunta ai metodi di base con `file` e `dns`.

- [Avahi](#) fornisce un'infrastruttura per il Multicast DNS Service Discovery in Debian.
- È equivalente a [Apple Bonjour / Apple Rendezvous](#).
- Il pacchetto plugin `libnss-mdns` fornisce la risoluzione dei nomi host attraverso mDNS per la funzionalità GNU Name Service Switch (NSS) della libreria C GNU (glibc).
- Il file `"/etc/nsswitch.conf"` dovrebbe contenere una sezione simile a `"hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4"`.
- I nomi host che terminano con lo [pseudo-dominio di livello più alto](#) (TLD) `".local"` vengono risolti.
- L'indirizzo multicast del collegamento locale mDNS IPv4 `"224.0.0.251"` o il suo equivalente IPv6 `"FF02::FB"` vengono usati per fare interrogazioni DNS per un nome che termina con `".local"`.

La risoluzione del nome host con il deprecato [NETBios su TCP/IP](#), usato dai vecchi sistemi Windows può essere fornita installando il pacchetto `winbind`. Per abilitare questa funzionalità il file `"/etc/nsswitch.conf"` dovrebbe avere una sezione come `"hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4 wins"`. (I sistemi Windows moderni solitamente usano il metodo `dns` per la risoluzione del nome host.)

Nota

L'[espansione dei gTLD \(generic Top-Level Domain, domini di più alto livello generici\)](#) nel [Domain Name System](#) (Sistema di nomi di dominio) è in corso d'opera. Fare attenzione a possibili [collisioni di nomi](#) quando si sceglie un nome di dominio usato solamente all'interno di una LAN.

5.1.2 Il nome dell'interfaccia di rete

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

5.1.3 L'intervallo degli indirizzi di rete per la LAN

Ecco un ripasso degli intervalli di indirizzi IPv4 a 32 bit in ciascuna classe, riservati per l'uso in [reti locali \(LAN\)](#) dalla [rfc1918](#). È garantito che questi indirizzi non creino conflitti con altri indirizzi della vera e propria rete Internet.

Nota

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

Classe	indirizzi di rete	maschera di rete	maschera rete /bit	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

Tabella 5.2: Elenco di intervalli di indirizzi di rete

Nota

Se uno di questi indirizzi viene assegnato ad un host, allora l'host non deve accedere ad Internet direttamente, ma deve accedervi attraverso un gateway che agisca da proxy per i singoli servizi o che, in alternativa, faccia da [NAT \(Network Address Translation, traduzione degli indirizzi di rete\)](#). I router a banda larga di solito agiscono da NAT per l'ambiente LAN dell'utente finale.

5.1.4 Il supporto per i dispositivi di rete

Nonostante la maggior parte dei dispositivi hardware sia supportata dal sistema Debian, ci sono alcuni dispositivi di rete che necessitano, per essere supportati, firmware non libero secondo la definizione nelle [DFSG](#). Vedere Sezione [9.10.5](#).

5.2 La configurazione moderna della rete per il desktop

Le interfacce di rete sono tipicamente inizializzate in "networking.service" per l'interfaccia `lo` e "NetworkManager.service" per le altre interfacce nei moderni sistemi desktop Debian che usano `systemd`.

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

- Questi sono forniti con propri programmi per l'interfaccia utente con [GUI](#) o a riga di comando.
- Sono forniti con un proprio [demone](#) come sistema di backend.
- Permettono al proprio sistema di connettersi facilmente ad Internet.
- Permettono una facile gestione della configurazione delle reti cablate e wireless.
- Permettono di configurare la rete in modo indipendente dal datato pacchetto `ifupdown`.

Nota

Non usare questi strumenti di configurazione automatica della rete per server. Sono pensati principalmente per utenti con desktop mobili su portatili.

Questi strumenti moderni di configurazione della rete devono essere configurati in modo corretto per evitare conflitti con il datato pacchetto `ifupdown` e con il suo file di configurazione `"/etc/network/interfaces"`.

5.2.1 Strumenti grafici di configurazione della rete

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

Fondamentalmente, la configurazione di rete per il desktop viene fatta nel modo seguente.

1. Aggiungere l'utente desktop, ad esempio `pippo`, al gruppo `netdev` con il comando seguente. (In alternativa farlo automaticamente attraverso [D-bus](#) nei moderni ambienti desktop come GNOME e KDE).

```
$ sudo adduser foo netdev
```

2. Mantenere la configurazione di `"/etc/network/interfaces"` semplice come la seguente.

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

3. Restart NM by the following.

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

4. Configurare la propria rete attraverso l'interfaccia grafica.

Nota

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

Suggerimento

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 La configurazione moderna della rete senza GUI

Con [systemd](#), la rete può essere configurata invece in `/etc/systemd/network/`. Vedere `systemd-resolved(8)`, `resolved.conf` e `systemd-networkd(8)`.

Ciò permette la configurazione moderna della rete senza GUI.

Una configurazione di client DHCP può essere impostata creando `"/etc/systemd/network/dhcp.network"`. Es.:

```
[Match]
Name=en*

[Network]
DHCP=yes
```

Una configurazione di rete statica può essere impostata creando `"/etc/systemd/network/static.network"`. Es.:

```
[Match]
Name=en*

[Network]
Address=192.168.0.15/24
Gateway=192.168.0.1
```

5.4 Configurazione della rete a basso livello

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

5.4.1 Comandi iproute2

I comandi [Iproute2](#) offrono funzionalità complete di configurazione della rete a basso livello. Quella che segue è una tabella di traduzione dai comandi obsoleti di [net-tools](#) ai nuovi comandi [iproute2](#) ecc.

net-tools obsoleti	nuovi iproute2, ecc.	manipolazione
<code>ifconfig(8)</code>	<code>ip addr</code>	indirizzo di protocollo (IP o IPv6) di un device
<code>route(8)</code>	<code>ip route</code>	voce nella tabella di instradamento
<code>arp(8)</code>	<code>ip neigh</code>	voce nella cache ARP o NDISC
<code>ipmaddr</code>	<code>ip maddr</code>	indirizzo multicast
<code>iptunnel</code>	<code>ip tunnel</code>	tunnel over IP
<code>nameif(8)</code>	<code>ifrename(8)</code>	nomina le interfacce di rete in base all'indirizzo MAC
<code>mii-tool(8)</code>	<code>ethtool(8)</code>	impostazioni del device Ethernet

Tabella 5.3: Tabella di traduzione dai comandi obsoleti `net-tools` ai nuovi comandi `iproute2`

Vedere `ip(8)` e [IPROUTE2 Utility Suite Howto](#).

5.4.2 Operazioni sicure a basso livello sulla rete

Si possono usare i comandi a basso livello per la rete seguenti in modo sicuro dato che non cambiano la configurazione della rete.

comando	descrizione
<code>ip addr show</code>	mostra lo stato del collegamento e l'indirizzo delle interfacce attive
<code>route -n</code>	mostra tutta la tabella di instradamento in indirizzi numerici
<code>ip route show</code>	mostra tutta la tabella di instradamento in indirizzi numerici
<code>arp</code>	mostra l'attuale contenuto delle tabelle cache ARP
<code>ip neigh</code>	mostra l'attuale contenuto delle tabelle cache ARP
<code>plog</code>	mostra il registro del demone ppp
<code>ping yahoo.com</code>	controlla la connessione Internet verso "yahoo.com"
<code>whois yahoo.com</code>	controlla chi ha registrato "yahoo.com" nel database dei domini
<code>traceroute yahoo.com</code>	traccia la connessione Internet verso "yahoo.com"
<code>tracert yahoo.com</code>	traccia la connessione Internet verso "yahoo.com"
<code>mtr yahoo.com</code>	traccia la connessione Internet verso "yahoo.com" (ripetutamente)
<code>dig [@dns-server.com] esempio.com [{a mx any}]</code>	controlla i record DNS di "esempio.com" con "dns-server.com" alla ricerca di un record "a", "mx" o "any"
<code>iptables -L -n</code>	controlla il filtraggio dei pacchetti
<code>netstat -a</code>	trova tutte le porte aperte
<code>netstat -l --inet</code>	trova le porte in ascolto
<code>netstat -ln --tcp</code>	trova le porte (numeriche) TCP in ascolto
<code>dlint esempio.com</code>	controlla le informazioni DNS di zona per "esempio.com"

Tabella 5.4: Elenco di comandi di rete a basso livello

Suggerimento

Alcuni di questi strumenti di configurazione di basso livello della rete sono contenuti in `/sbin/`. Potrebbe essere necessario fornire il percorso completo dei comandi, come `/sbin/ifconfig` o aggiungere `/sbin` all'elenco in `PATH` nel proprio file `~/ .bashrc`.

5.5 Ottimizzazione della rete

L'ottimizzazione generale della rete va oltre gli scopi che questo documento si prefigge. Vengono trattati solo quegli argomenti che interessano le connessioni dei computer personali.

pacchetto	popcon	dimensione	descrizione
iftop	V:8, I:118	97	mostra informazioni sull'uso della banda su di un'interfaccia di rete
iperf	V:4, I:55	305	strumento di misurazione della banda del protocollo Internet
ifstat	V:0, I:8	60	Monitoraggio delle STATistiche sulle InterFacce
bmon	V:1, I:18	146	strumento portatile per monitoraggio della banda e stime della velocità
ethstatus	V:0, I:4	40	script che misura velocemente la quantità di dati passanti per un device di rete
bing	V:0, I:1	80	tester empirico stocastico della banda
bwm-ng	V:2, I:18	90	strumento piccolo e semplice basato sulla console per il monitoraggio della banda
ethstats	V:0, I:0	23	strumento basato su console per il monitoraggio delle statistiche Ethernet
ipfm	V:0, I:0	78	strumento di analisi della banda

Tabella 5.5: Elenco degli strumenti di ottimizzazione della rete

5.5.1 Trovare l'MTU ottimale

Il valore di **MTU** ([Maximum Transmission Unit, unità massima di trasmissione](#)) può essere determinato sperimentalmente con `ping(8)` usando l'opzione `-M do` che invia pacchetti ICMP con una dimensione dei dati a partire da 1500 (con un offset di 28 byte per l'intestazione IP+ICMP) e trova la dimensione maggiore che non comporta frammentazione IP.

Per esempio, provare a fare quanto segue.

```
$ 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.

Questo processo si chiama [Path MTU \(PMTU\) discovery \(RFC1191\)](#) e il comando `tracert(8)` può automatizzarlo.

Suggerimento

L'esempio precedente di PMTU con un valore di 1454 è per il mio precedente fornitore FTTP che usava [ATM \(Asynchronous Transfer Mode\)](#) come rete backbone e forniva ai clienti un servizio [PPPoE](#). Il valore effettivo di PMTU dipende dal proprio ambiente, ad esempio è 1500 per il mio nuovo fornitore FTTP.

ambiente di rete	MTU	motivo
collegamento dial-up (IP: PPP)	576	standard
collegamento Ethernet (IP: DHCP o statico)	1500	standard e predefinito
collegamento Ethernet (IP: PPPoE)	1492 (=1500-8)	2 byte per l'intestazione PPP e 6 byte per l'intestazione PPPoE
collegamento Ethernet (backbone del fornitore servizi: ATM, IP: DHCP o statico)	1462 (=48*31-18-8)	ipotesi dell'autore: 18 byte per l'intestazione Ethernet, 8 byte per la coda SAR
collegamento Ethernet (backbone del fornitore servizi: 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

Tabella 5.6: Linee guida di base per il valore di MTU ottimale

In aggiunta a queste linee guida di base si dovrebbe sapere quanto segue.

- Qualsiasi uso di metodi di tunnelling (VPN, ecc.) può ridurre ulteriormente l'MTU ottimale per il loro carico.
- Il valore di MTU non dovrebbe superare il valore di PMTU determinato sperimentalmente.
- A patto di seguire le indicazioni precedenti un valore maggiore di MTU è meglio.

Il valore di **MSS (Maximum Segment Size, dimensione massima di segmento)**MSS è usato come misura alternativa della dimensione dei pacchetti. Le relazioni tra MSS e MTU sono le seguenti.

- MSS = MTU - 40 per IPv4
- MSS = MTU - 60 per IPv6

Nota

L'ottimizzazione basata su `iptables(8)` (vedere Sezione 5.6) può bloccare la dimensione dei pacchetti al valore MSS ed è utile per il router. Vedere "TCP MSS" in `iptables(8)`.

5.5.2 Ottimizzazione di TCP per la WAN

Il flusso TCP può essere massimizzato regolando i parametri di dimensione del buffer TCP come descritto in "TCP Tuning Guide" e "TCP tuning" per le moderne WAN a larga banda e alta latenza. Per adesso, le attuali impostazioni predefinite di Debian funzionano bene persino per la mia LAN connessa con un servizio FTTP veloce a 1Gbps.

5.6 Infrastruttura netfilter

Netfilter fornisce un'infrastruttura per **firewall con stati** e **traduzione di indirizzi di rete (NAT)** con moduli del **kernel Linux** (vedere Sezione 3.8.1).

Il principale programma in spazio utente di **netfilter** è `iptables(8)`. Si può configurare manualmente **netfilter** in modo interattivo dalla shell, salvare il suo stato con `iptables - save(8)` e ripristinarlo attraverso uno script di init con `iptables - restore(8)` al riavvio del sistema.

Script di aiuto per la configurazione come **shorewall** facilitano questo processo.

Vedere la documentazione all'indirizzo <http://www.netfilter.org/documentation/> (o in `/usr/share/doc/iptables/html/`).

- [Linux Networking-concepts HOWTO](#)
-

pacchetto	popcon	dimensione	descrizione
iptables	V:314, I:990	2521	strumenti di amministrazione per netfilter (iptables(8) per IPv4, ip6tables(8) per IPv6)
arptables	V:0, I:2	96	strumenti di amministrazione per netfilter (arptables(8) per ARP)
ebtables	V:16, I:37	265	strumenti di amministrazione per netfilter (ebtables(8) per bridge Ethernet)
iptstate	V:0, I:3	116	monitora continuamente lo stato di netfilter (in modo simile a top(1))
shorewall-init	V:0, I:0	68	inizializzazione di Shoreline Firewall
shorewall	V:4, I:12	2458	generatore di file di configurazione per Shoreline Firewall , netfilter
shorewall-lite	V:0, I:0	65	generatore di file di configurazione per Shoreline Firewall , netfilter (versione leggera)
shorewall6	V:1, I:2	779	generatore di file di configurazione per Shoreline Firewall , netfilter (versione IPv6)
shorewall6-lite	V:0, I:0	64	generatore di file di configurazione per Shoreline Firewall , netfilter (versione IPv6 leggera)

Tabella 5.7: Elenco di strumenti per firewall

- [Linux 2.4 Packet Filtering HOWTO](#)
- [Linux 2.4 NAT HOWTO](#)

Suggerimento

Anche se questi documenti sono stati scritti per Linux **2.4**, sia il comando `iptables(8)` sia la funzione `netfilter` del kernel sono validi per gli attuali kernel Linux **2.6** e **3.x**.

Capitolo 6

Applicazioni per la rete

Dopo aver stabilito la connessione di rete (vedere Capitolo 5), si possono eseguire svariate applicazioni per la rete.

Suggerimento

Per una guida moderna sull'infrastruttura di rete specifica per Debian leggere [The Debian Administrator's Handbook —Network Infrastructure](#).

Suggerimento

Se si abilita la "Verifica in 2 passaggi" con alcuni ISP è necessario ottenere una password di applicazione per accedere ai servizi POP e SMTP dal proprio programma. Può essere necessario approvare l'IP del proprio host in anticipo.

6.1 Browser web

Ci sono molti pacchetti per [browser web](#) per accedere a contenuti remoti tramite [HTTP \(Hypertext Transfer Protocol\)](#).

pacchetto	popcon	dimensioni	tipo	descrizione del browser web
chromium	V:54, I:134	192549	per X	Chromium , (browser open-source da Google)
firefox	V:13, I:20	209827	” ”	Firefox , (browser open-source da Mozilla, disponibile solo in Debian Unstable)
firefox-esr	V:199, I:421	200011	” ”	Firefox ESR , (Firefox Extended Support Release)
epiphany-browser	V:4, I:23	3730	” ”	Epiphany , per GNOME , aderente alle HIG
konqueror	V:17, I:95	20763	” ”	Konqueror , per KDE
dillo	V:1, I:7	1536	” ”	Dillo , (browser leggero, basato su FLTK)
w3m	V:25, I:252	2367	testuali	w3m
lynx	V:11, I:97	1948	” ”	Lynx
elinks	V:5, I:27	1767	” ”	ELinks
links	V:5, I:38	2250	” ”	Links (solo testo)
links2	V:1, I:14	5418	grafici	Links (grafica in console senza X)

Tabella 6.1: Elenco di browser web

6.1.1 Configurazione del browser

Per alcuni browser si può accedere alle loro impostazioni usando le seguenti stringhe URL speciali.

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

Debian offre, nell'area main dell'archivio, molti pacchetti liberi con plugin per browser che possono gestire non soltanto [Java](#) (piattaforma software) e [Flash](#), ma anche file [MPEG](#), [MPEG2](#), [MPEG4](#), [DivX](#), [Windows Media Video \(.wmv\)](#), [QuickTime \(.mov\)](#), [MP3 \(.mp3\)](#), file [Ogg/Vorbis](#), DVD, VCD, ecc. Debian offre anche, nell'area contrib o non-free dell'archivio, programmi di aiuto per installare pacchetti con plugin non liberi per il browser.

pacchetto	popcon	dimensione	area	descrizione
pepperflashplugin-nonfree	V:0, I:15	26	contrib	Pepper Flash Player - plugin per browser

Tabella 6.2: Elenco di pacchetti con plugin per browser

Suggerimento

Benché l'uso dei pacchetti Debian elencati sopra sia molto più facile, è sempre possibile abilitare i plugin manualmente installando i file "*.so" nelle directory dei plugin (ad esempio, "/usr/lib/iceweasel/plugins/") e riavviando il browser.

Alcuni siti web rifiutano la connessione in base alla stringa user-agent del browser. Si può aggirare questa situazione [mimando la stringa user-agent](#). Per esempio, può essere fatto aggiungendo nei file di configurazione, come "~/.gnome2/epiphany/mozilla/epiphany" o "~/.mozilla/firefox/*.default/user.js", la riga seguente.

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

In alternativa si può aggiungere e reimpostare questa variabile digitando "about:config" al posto dell'URL e cliccando con il pulsante destro sul suo contenuto.



Attenzione

La stringa user-agent falsificata può causare dei [brutti effetti collaterali con Java](#).

6.2 Il sistema di posta

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



Attenzione

Se si desidera impostare il server di posta per scambiare la posta direttamente con Internet, si dovrebbe leggere una documentazione più dettagliata di questa documentazione base.

6.2.1 Nozioni di base sulla posta elettronica

Un messaggio [email](#) è formato da tre componenti: la busta del messaggio, l'intestazione del messaggio e il corpo del messaggio.

- Le informazioni "To" e "From" nella busta del messaggio vengono usate dall'[SMTP](#) per consegnare l'email. (L'informazione "From" nella busta del messaggio è anche chiamata [indirizzo di bounce](#), From_, ecc.)
- Le informazioni "To" e "From" nell'intestazione del messaggio vengono visualizzate dal [programma di posta](#). (Benché nella maggior parte dei casi questi sono identici a quelli nella busta del messaggio, può non essere sempre così.)
- 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.

pacchetto	popcon	dimensione	tipo
evolution	V:29, I:219	475	programma per X con interfaccia utente grafica (GNOME3, suite groupware)
thunderbird	V:57, I:134	168690	X GUI program (GTK, Mozilla Thunderbird)
kmail	V:33, I:85	18011	programma per X con interfaccia utente grafica (KDE)
mutt	V:33, I:278	7071	programma basato su terminale a caratteri probabilmente usato con vim
mew	V:0, I:0	2334	programma basato su terminale a caratteri in (x) emacs

Tabella 6.3: Elenco di programmi di posta (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.
- Non è realistico attendersi che un singolo smarthost invii messaggi di indirizzi di posta di origine non correlati ad host remoti in modo affidabile.

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).
- [Tecniche anti-spam](#) come [DKIM \(DomainKeys Identified Mail\)](#) e [SPF \(Sender_Policy_Framework\)](#) e [DMARC \(Domain-based Message Authentication, Reporting and Conformance\)](#) sono usate di frequente per il [filtraggio delle email](#).
- Il servizio [DomainKeys Identified Mail](#) può essere fornito per la propria posta inviata attraverso uno 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 di trasporto della posta (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 Sezione 6.2.2 and Sezione 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`.

Suggerimento

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 La configurazione di `exim4`

Per la posta di Internet attraverso uno smarthost, riconfigurare i pacchetti `exim4-*` nel modo seguente.

```
$ sudo systemctl stop exim4
$ sudo dpkg-reconfigure exim4-config
```

pacchetto	popcon	dimensione	descrizione
exim4-daemon-light	V:320, I:346	1517	agente di trasporto della posta Exim4 (MTA predefinito in Debian)
exim4-daemon-heavy	V:7, I:8	1668	Exim4 mail transport agent (MTA: flexible alternative)
exim4-base	V:328, I:355	1720	documentazione (formato testo) e file comuni per Exim4
exim4-doc-html	I:1	3667	documentazione per Exim4 (in formato HTML)
exim4-doc-info	I:1	625	documentazione per Exim4 (in formato info)
postfix	V:147, I:162	4183	Postfix mail transport agent (MTA: secure alternative)
postfix-doc	I:9	4444	documentazione per Postfix (formati HTML e testo)
sasl2-bin	V:5, I:19	428	implementazione dell'API SASL Cyrus (di supporto a Postfix per SMTP AUTH)
cyrus-sasl2-doc	I:1	575	SASL Cyrus - documentazione
msmtp	V:6, I:12	547	Light weight MTA
msmtp-mta	V:4, I:5	86	Light weight MTA (sendmail compatibility extension to <code>msmtp</code>)
esmtp	V:0, I:0	128	Light weight MTA
esmtp-run	V:0, I:0	32	Light weight MTA (sendmail compatibility extension to <code>esmtp</code>)
nullmailer	V:8, I:10	479	Strip down MTA, no local mail
ssmtp	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.)

Tabella 6.4: List of basic mail transport agent related packages

Selezionare "posta inviata tramite «uno smarthost», ricevuta via SMTP o fetchmail" per "Tipo di configurazione del sistema di posta".

Impostare "Mail name del sistema:" al suo valore predefinito come FQDN (vedere Sezione 5.1.1).

Impostare "indirizzi IP sui quali attendere connessioni SMTP in ingresso:" al suo valore predefinito "127.0.0.1 ; ::1".

Svuotare il contenuto di "Altre destinazioni per conto delle quali accettare posta:".

Svuotare il contenuto di "Sistemi per i quali fare il "relay":".

Impostare l'"Indirizzo IP o nome host dello smarthost per la posta in uscita:" a "smtp.hostname.dom:587".

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

Rispondere a "Mantenere al minimo il numero di richieste DNS (Dial-on-Demand)?" in uno dei modi seguenti.

- "No", se il sistema è connesso ad Internet durante l'avvio.
- "Sì", se il sistema **non** è connesso ad Internet durante l'avvio.

Impostare "Modalità di consegna per la posta locale:" a "Formato mbox in /var/mail/".

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

Creare voci per la password dello smarthost modificando il file `/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)

Avviare `exim4` con il comando seguente.

```
$ sudo systemctl start exim4
```

Il nome host in `"/etc/exim4/passwd.client"` non dovrebbe essere l'alias. Si può controllare il vero nome host nel modo seguente.

```
$ host smtp.hostname.dom
smtp.hostname.dom is an alias for smtp99.hostname.dom.
smtp99.hostname.dom has address 123.234.123.89
```

Per aggirare il problema degli alias, io uso espressioni regolari nel file `"/etc/exim4/passwd.client"` SMTP AUTH probabilmente funziona anche se il fornitore di servizi Internet sposta l'host a cui punta l'alias.

Si può aggiornare manualmente la configurazione di `exim4` facendo quanto segue:

- Aggiornare i file di configurazione di `exim4` in `"/etc/exim4/"`.
 - Creare `"/etc/exim4/exim4.conf.localmacros"` per impostare le MACRO e modificare `"/etc/exim4/exim4.conf"` (Configurazione non suddivisa.)
 - Creare nuovi file o modificare quelli esistenti nelle sottodirectory `"/etc/exim4/exim4.conf.d"`. (Configurazione suddivisa.)
- Run `"systemctl reload exim4"`.



Attenzione

Se si è scelta la risposta "No" (risposta predefinita) per la domanda di `debconf` "Mantenere al minimo il numero di richieste DNS (Dial-on-Demand)?" ed il sistema **non** è connesso ad Internet, l'avvio di `exim4` può richiedere un tempo molto lungo.

Leggere la guida ufficiale in `"/usr/share/doc/exim4-base/README.Debian.gz"` e `update-exim4.conf(8)`.



avvertimento

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 La configurazione di Postfix con SASL

Per la posta Internet via smarthost, si dovrebbe come prima cosa leggere la [documentazione di Postfix](#) e le pagine man principali.

comando	funzione
<code>postfix(1)</code>	programma di controllo di Postfix
<code>postconf(1)</code>	utilità di configurazione per Postfix
<code>postconf(5)</code>	parametri di configurazione di Postfix
<code>postmap(1)</code>	gestione delle tabelle di consultazione di Postfix
<code>postalias(1)</code>	gestione del database degli alias di Postfix

Tabella 6.5: Elenco delle pagine di manuale di Postfix importanti

Si possono (ri)configurare i pacchetti `postfix` e `sasl2-bin` nel modo seguente.

```
$ sudo systemctl stop postfix
$ sudo dpkg-reconfigure postfix
```

Scegliere "Internet con smarthost".

Impostare "relay host SMTP (vuoto per nessuno):" a "[smtp.hostname.dom]:587" e configurarlo nel modo seguente.

```
$ 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
```

Creare le voci con le password per lo smarthost.

```
$ cat /etc/postfix/sasl_passwd
[smtp.hostname.dom]:587      username:password
$ sudo postmap hash:/etc/postfix/sasl_passwd
```

Avviare `postfix` con il comando seguente.

```
$ sudo systemctl start postfix
```

In questo caso l'uso di "[]" nel dialogo di `dpkg-reconfigure` e in `/etc/postfix/sasl_passwd` assicura che non venga controllato il record MX, ma usato direttamente l'esatto nome host specificato. Vedere "Abilitare l'autenticazione SASL nel client SMTP di Postfix" in `/usr/share/doc/postfix/html/SASL_README.html`.

6.2.4.3 La configurazione dell'indirizzo di posta

Ci sono diversi [file di configurazione dell'indirizzo di posta per gli strumenti di trasporto della posta, di consegna e per i programmi di posta](#).

file	funzione	applicazione
<code>/etc/mailname</code>	nome host predefinito per la posta (in uscita)	specifico di Debian, <code>mailname(5)</code>
<code>/etc/email-addresses</code>	sostituzione del home host per la posta in uscita	specifico di <code>exim(8)</code> , <code>exim4-config_files(5)</code>
<code>/etc/postfix/generic</code>	sostituzione del home host per la posta in uscita	specifico di <code>postfix(1)</code> , attivato dopo l'esecuzione del comando <code>postmap(1)</code>
<code>/etc/aliases</code>	alias dei nomi degli account per la posta in entrata	generale, attivato dopo l'esecuzione del comando <code>newaliases(1)</code>

Tabella 6.6: Elenco dei file di configurazione correlati all'indirizzo di posta

Il **nomeposta** nel file `/etc/mailname` è solitamente un nome di dominio pienamente qualificato (FQDN) che è risolto ad uno degli indirizzi IP dell'host. Per le postazioni mobili che non hanno un nome host con un indirizzo IP risolvibile, impostare questo **nomemail** al valore di `hostname -f`. (Questa è una scelta sicura e funziona sia per `exim4` - * sia per `postfix`.)

Suggerimento

Il contenuto di `/etc/mailname` è usato da molti programmi non MTA per determinare il loro comportamento predefinito. Per `mutt`, impostare le variabili `hostname` e `from` nel file `~/muttrc` per sovrascrivere il valore di **mailname**. Per i programmi nel pacchetto `devscripts`, come `bts(1)` e `dch(1)`, esportare le variabili d'ambiente `"$DEBFULLNAME"` e `"$DEBEMAIL"` per sovrascriverlo.

Suggerimento

Il pacchetto `popularity-contest` normalmente invia posta dall'account di `root` con FQDN. È necessario impostare `MAILFROM` in `/etc/popularity-contest.conf` come descritto nel file `/usr/share/popularity-contest/default.conf`. In caso contrario la posta verrà rigettata dal server SMTP dello smarthost. Sebbene sia una scocciatura, questo approccio è più sicuro della riscrittura dell'indirizzo sorgente di tutta la posta di `root` da parte dell'MTA e dovrebbe essere usato per altri demoni e script cron.

Quando si imposta **mailname** al valore di "hostname -f", la modifica dell'indirizzo di posta d'origine con il MTA può essere fatta nel modo seguente.

- Usando il file "/etc/email-addresses" per `exim4(8)`, come spiegato in `exim4-config_files(5)`.
- Usando il file "/etc/postfix/generic" per `postfix(8)`, come spiegato in `generic(5)`.

Per `postfix` sono necessarie, in aggiunta, le azioni seguenti.

```
# postmap hash:/etc/postfix/generic
# postconf -e 'smtp_generic_maps = hash:/etc/postfix/generic'
# postfix reload
```

Si può testare la configurazione dell'indirizzo di posta nel modo seguente.

- per `exim(8)` con le opzioni `-brw`, `-bf`, `-bF`, `-bV`, ...
- per `postmap(1)` con l'opzione `-q`.

Suggerimento

Exim viene fornito con svariati programmi di utilità, come `exiqgrep(8)` e `exipick(8)`. Per conoscere i comandi disponibili vedere "dpkg -L exim4-base|grep man8/".

6.2.4.4 Operazioni di base degli MTA

Ci sono svariate operazioni di base degli MTA. Alcune possono essere fatte usando l'interfaccia di compatibilità con `sendmail(1)`.

comando exim	comando postfix	descrizione
<code>sendmail</code>	<code>sendmail</code>	legge la posta dallo standard input e gestisce la consegna (<code>-bm</code>)
<code>mailq</code>	<code>mailq</code>	elenca la coda della posta con stato e ID dei messaggi in coda (<code>-bp</code>)
<code>newaliases</code>	<code>newaliases</code>	inizializza il database degli alias (<code>-I</code>)
<code>exim4 -q</code>	<code>postqueue -f</code>	invia tutta la posta in attesa (<code>-q</code>)
<code>exim4 -qf</code>	<code>postsuper -r ALL deferred; postqueue -f</code>	invia tutta la posta
<code>exim4 -qff</code>	<code>postsuper -r ALL; postqueue -f</code>	invia anche la posta congelata
<code>exim4 -Mg id_in_coda</code>	<code>postsuper -h id_in_coda</code>	congela un messaggio in base al suo ID nella coda
<code>exim4 -Mrm id_in_coda</code>	<code>postsuper -d id_in_coda</code>	rimuove un messaggio in base al suo ID nella coda
N/D	<code>postsuper -d ALL</code>	rimuove tutti i messaggi

Tabella 6.7: Elenco di operazioni base degli MTA

Suggerimento

Può essere una buona idea far inviare tutta la posta da uno script in "/etc/ppp/ip-up.d/*".

6.3 Il server e le utilità per l'accesso remoto (SSH)

SSH (Secure SHell) è il metodo **sicuro** per connettersi in Internet. Una versione libera di SSH chiamata [OpenSSH](#) è disponibile nei pacchetti Debian `openssh-client` e `openssh-server`.

Per l'utente `ssh(1)` funziona come un `telnet(1)` più intelligente e più sicuro. A differenza del comando `telnet`, `ssh` non si interrompe a seguito del carattere di escape di `telnet` (impostazione predefinita iniziale CTRL-`]`).

pacchetto	popcon	dimensione	strumento	descrizione
openssh-client	V:809, I:997	4298	<code>ssh(1)</code>	client SSH
openssh-server	V:701, I:844	1567	<code>sshd(8)</code>	server SSH
ssh-askpass	V:2, I:32	106	<code>ssh-askpass(1)</code>	chiede all'utente una passphrase per <code>ssh-add</code> (X semplice)
ssh-askpass-gnome	V:0, I:5	294	<code>ssh-askpass-gnome(1)</code>	askpass (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

Tabella 6.8: Elenco dei server e delle utilità per l'accesso remoto

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

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



Attenzione

Se il proprio SSH è accessibile da Internet, vedere Sezione [4.6.3](#).

Suggerimento

Per permettere al processo della shell remota di sopravvivere all'interruzione della connessione, usare il programma `screen(1)` (vedere Sezione [9.1.2](#)).

6.3.1 Nozioni di base su 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)`.



avvertimento

Se si desidera eseguire il server OpenSSH, non deve esistere il file `"/etc/ssh/sshd_not_to_be_run"`. Don't enable rhost based authentication (`HostbasedAuthentication` in `/etc/ssh/sshd_config`).

I comandi seguenti avviano una connessione `ssh(1)` da un client.

file di configurazione	descrizione del file di configurazione
/etc/ssh/ssh_config	impostazioni predefinite per il client SSH, vedere <code>ssh_config(5)</code>
/etc/ssh/sshd_config	impostazioni predefinite per il server SSH, vedere <code>ssh_config(5)</code>
~/.ssh/authorized_keys	chiavi SSH pubbliche predefinite che i client usano per connettersi a questo account su questo server SSH
~/.ssh/id_rsa	chiave RSA SSH-2 segreta dell'utente
~/.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

Tabella 6.9: Elenco dei file di configurazione per SSH

comando	descrizione
<code>ssh nomeutente@nomehost.dominio.ext</code>	connette nella modalità predefinita
<code>ssh -v nomeutente@nomehost.dominio.ext</code>	connette nella modalità predefinita con messaggi di debug
<code>ssh -o PreferredAuthentications=password username@hostname.domain.ext</code>	forza l'uso di password con SSH versione 2
<code>ssh -t username@hostname.domain.ext passwd</code>	run <code>passwd</code> program to update password on a remote host

Tabella 6.10: Elenco di esempi di avvio di client 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 Connettersi senza password remote

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".

Generare localmente le chiavi di autenticazione ed installare la chiave pubblica sul sistema remoto con i comandi seguenti.

```
$ 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 Gestire client SSH estranei

Sono disponibili alcuni client [SSH](#) liberi per altre piattaforme.

ambiente	programma ssh libero
Windows	puTTY (http://www.chiark.greenend.org.uk/~sgtatham/putty/) (GPL)
Windows (cygwin)	SSH in cygwin (http://www.cygwin.com/) (GPL)
Macintosh Classic	macSSH (http://www.macssh.com/) (GPL)
Mac OS X	OpenSSH; usare ssh nell'applicazione Terminale (GPL)

Tabella 6.11: Elenco di client SSH per altre piattaforme

6.3.5 Impostare ssh-agent

È meglio per ragioni di sicurezza proteggere la propria chiave segreta di autenticazione SSH con una passphrase. Se non è già stata impostata una passphrase usare "ssh-keygen -p" per farlo.

Mettere la propria chiave SSH pubblica (ad esempio "~/.ssh/id_rsa.pub") in "~/.ssh/authorized_keys" su un host remoto usando una connessione all'host remoto basata su password, come descritto in precedenza.

```
$ 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)
```

Per il comando successivo non sarà più necessaria da questo momento la password remota.

```
$ scp foo username@remote.host:foo
```

Premere ~D per terminare la sessione ssh-agent.

Per il server X, il normale script di avvio Debian esegue ssh-agent come processo genitore. Perciò è necessario eseguire ssh-add una volta sola. Per ulteriori informazioni, leggere 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 Forwarding della porta per tunnel SMTP/POP3

Per stabilire una pipe per connettersi alla porta 25 del server-remoto dalla porta 4025 dell'host locale e alla porta 110 del server-remoto dalla porta 4110 dell'host locale attraverso ssh, eseguire sull'host locale il comando seguente.

```
# ssh -q -L 4025:remote-server:25 4110:remote-server:110 username@remote-server
```

Questo è un metodo sicuro di creare connessioni a server SMTP/POP3 in Internet. Impostare nel file "/etc/ssh/sshd_config" dell'host remoto la voce "AllowTcpForwarding" a "yes".

6.3.8 Spegnerne il sistema remoto su SSH

È necessario proteggere il processo che esegue "shutdown -h now" (vedere Sezione 1.1.8) dalla terminazione di SSH usando il comando at(1) (vedere Sezione 9.4.13) nel modo seguente.

```
# echo "shutdown -h now" | at now
```

Eseguire "shutdown -h now" in una sessione screen(1) (vedere Sezione 9.1.2) è un altro modo di ottenere lo stesso risultato.

6.3.9 Risolvere il problemi con SSH

Se si hanno problemi, controllare i permessi dei file di configurazione ed eseguire ssh con l'opzione "-v".

Se si è root e si hanno problemi con un firewall usare l'opzione "-p"; questo evita l'uso delle porte 1 - 1023 del server.

Se le connessioni ssh ad un sito remoto smettono di funzionare improvvisamente, potrebbe essere a causa di modifiche fatte dall'amministratore di sistema, molto probabilmente cambiamenti in "host - key" durante l'amministrazione del sistema. Dopo essersi accertati che questa sia davvero la causa e che nessuna stia cercando di falsificare l'host remoto con qualche truccetto, si può riottenere una connessione rimuovendo la voce "host - key" dal file "~/.ssh/known_hosts" sull'host locale.

6.4 Server ed utilità per la stampa

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 Sezione [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.

Grazie alla funzionalità di auto-conversione in base al formato dei file del sistema CUPS, il semplice passaggio di qualsiasi tipo di dati al comando `lpr` dovrebbe generare l'output di stampa atteso. (In CUPS, il comando `lpr` può essere abilitato installando il pacchetto `cups-bsd`.)

Il sistema Debian ha alcuni pacchetti degni di nota per ciò che riguarda i server e le utilità di stampa

pacchetto	popcon	dimensione	porta	descrizione
lpr	V:3, I:4	362	printer (515)	lpr/lpd BSD (Line printer daemon) , demone di stampa in linea)
lprng	V:0, I:1	3064	" "	" " (Migliorato)
cups	V:129, I:376	1134	IPP (631)	server CUPS di stampa Internet
cups-client	V:139, I:434	493	" "	comandi System V per la stampa con CUPS: <code>lp(1)</code> , <code>lpstat(1)</code> , <code>lptions(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	" "	comandi BSD per la stampa con 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	Non applicabile	driver di stampa per CUPS

Tabella 6.12: Elenco di server e utilità di stampa

Suggerimento

Si può configurare il sistema CUPS indirizzando il proprio browser web all'indirizzo "<http://localhost:631/>".

6.5 Altri server di rete

Ecco un elenco di altri server applicativi di rete.

CIFS (Common Internet File System Protocol) è lo stesso protocollo di [SMB \(Server Message Block\)](#) ed è ampiamente usato da Microsoft Windows.

pacchetto	popcon	dimensione	protocollo	descrizione
telnetd	V:1, I:2	111	TELNET	server TELNET
telnetd-ssl	V:0, I:0	169	” ”	” ” (supporto SSL)
nfs-kernel-server	V:42, I:78	342	NFS	condivisione di file Unix
samba	V:105, I:155	16630	SMB	condivisione di file e stampanti Windows
netatalk	V:2, I:3	2077	ATP	condivisione di file e stampanti Apple/Mac (AppleTalk)
proftpd-basic	V:23, I:31	488	FTP	scaricamento generico di file
apache2	V:244, I:310	614	HTTP	server web generico
squid	V:12, I:14	8381	” ”	server proxy web generico
squid3	V:3, I:9	240	” ”	” ”
bind9	V:52, I:63	1070	DNS	indirizzo IP per altri host
isc-dhcp-server	V:18, I:52	1471	DHCP	indirizzo IP per il client stesso

Tabella 6.13: Elenco di altri server applicativi di rete

Suggerimento

Vedere Sezione [4.5.2](#) per l'integrazione di sistemi server.

Suggerimento

La risoluzione del nome di host è solitamente fornita dal server [DNS](#). Per l'indirizzo IP dell'host assegnato dinamicamente da [DHCP](#), può essere impostato un [DNS dinamico](#) per la risoluzione del nome host, usando [bind9](#) e [isc-dhcp-server](#) come descritto nella [pagina del Wiki Debian sui DDNS](#).

Suggerimento

L'uso di server proxy come [squid](#) è molti più efficiente per risparmiare banda rispetto all'uso di server mirror locali con il completo contenuto dell'archivio Debian.

6.6 Altri client di rete

Ecco un elenco di altri client applicativi di rete.

6.7 Diagnosi dei demoni di sistema

Il programma `telnet` permette la connessione manuale ai demoni di sistema e la loro diagnosi.

Per testare il semplice servizio [POP3](#) provare il comando seguente.

```
$ telnet mail.ispname.net pop3
```

Per testare il servizio [POP3](#) con [TLS/SSL](#) abilitato di alcuni fornitori di servizi Internet, è necessario un client `telnet` con [TLS/SSL](#) abilitato fornito dal pacchetto `telnet-ssl` o `openssl`.

```
$ telnet -z ssl pop.gmail.com 995
```

```
$ openssl s_client -connect pop.gmail.com:995
```

Le seguenti [RFC](#) forniscono le conoscenze necessarie per ciascun demone di sistema.

L'uso delle porte è descritto in `"/etc/services"`.

pacchetto	popcon	dimensione	protocollo	descrizione
netcat	I:43	16	TCP/IP	coltellino svizzero TCP/IP
openssl	V:814, I:994	1465	SSL	eseguibile SSL (Secure Socket Layer) e strumenti crittografici relativi
stunnel4	V:6, I:16	520	” ”	wrapper SSL universale
telnet	V:61, I:901	163	TELNET	client TELNET
telnet-ssl	V:0, I:3	209	” ”	” ” (supporto SSL)
nfs-common	V:182, I:323	768	NFS	condivisione di file Unix
smbclient	V:16, I:174	2017	SMB	client per condivisione di file e stampanti MS Windows
cifs-utils	V:29, I:123	307	” ”	comandi mount e umount per file MS Windows remoti
ftp	V:16, I:251	137	FTP	client FTP
lftp	V:6, I:40	2255	” ”	” ”
ncftp	V:3, I:21	1339	” ”	client FTP a tutto schermo
wget	V:255, I:987	3477	HTTP e FTP	scaricatore per web
curl	V:160, I:575	427	” ”	” ”
axel	V:0, I:4	216	” ”	scaricatore accelerato
aria2	V:2, I:19	1854	” ”	scaricatore accelerato con gestione di BitTorrent e Metalink
bind9-host	V:176, I:946	370	DNS	host(1) da bind9, "Priorità: standard"
dnsutils	V:59, I:490	261	” ”	dig(1) da bind, "Priorità: standard"
isc-dhcp-client	V:222, I:980	686	DHCP	ottiene indirizzo IP
ldap-utils	V:12, I:74	719	LDAP	ottiene dati da server LDAP

Tabella 6.14: Elenco di altri client applicativi di rete

RFC	descrizione
rfc1939 e rfc2449	servizio POP3
rfc3501	servizio IMAP4
rfc2821 (rfc821)	servizio SMTP
rfc2822 (rfc822)	formato file di posta
rfc2045	MIME (Multipurpose Internet Mail Extensions)
rfc819	servizio DNS
rfc2616	servizio HTTP
rfc2396	definizione di URI

Tabella 6.15: Elenco di RFC popolari

Capitolo 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	dimensione	descrizione
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

Tabella 7.1: List of desktop environment

Suggerimento

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 Sezione [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 "[modedb 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.

pacchetto	popcon	dimensione del pacchetto	descrizione
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

Tabella 7.2: List of notable GUI infrastructure packages

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

Suggerimento

`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.

pacchetto	popcon	dimensione del pacchetto	tipo	descrizione
evolution	V:29, I:219	475	GNOME	gestione di informazioni personali (groupware e posta elettronica)
thunderbird	V:57, I:134	168690	GTK	Email client (Mozilla Thunderbird)
kontakt	V:1, I:14	2152	KDE	gestione di informazioni personali (groupware e posta elettronica)
libreoffice-writer	V:167, I:423	39325	LO	elaboratore di testi
abiword	V:1, I:11	5133	GNOME	elaboratore di testi
calligrawords	V:0, I:7	5717	KDE	elaboratore di testi
scribus	V:1, I:22	30370	KDE	desktop publishing editor to edit PDF files
glabels	V:0, I:4	1326	GNOME	editor di etichette
libreoffice-calc	V:166, I:418	32969	LO	foglio di calcolo
gnumeric	V:5, I:20	9933	GNOME	foglio di calcolo
calligrasheets	V:0, I:5	10890	KDE	foglio di calcolo
libreoffice-impress	V:149, I:415	9930	LO	presentazioni
calligrastage	V:0, I:5	5102	KDE	presentazioni
libreoffice-base	V:116, I:289	7465	LO	gestione di database
kexi	V:0, I:2	7576	KDE	gestione di database
libreoffice-draw	V:150, I:416	14596	LO	editor di grafica vettoriale (draw)
inkscape	V:42, I:188	84892	GNOME	editor di grafica vettoriale (draw)
karbon	V:0, I:6	3473	KDE	editor di grafica vettoriale (draw)
dia	V:4, I:30	3727	GTK	editor di grafi e diagrammi di flusso
gimp	V:58, I:315	19706	GTK	editor di grafica bitmap (paint)
shotwell	V:16, I:217	6451	GTK	organizzatore di foto digitali
digikam	V:2, I:12	2646	KDE	organizzatore di foto digitali
darktable	V:6, I:17	22902	GTK	lighttable and darkroom for photographers
planner	V:0, I:4	1146	GNOME	gestione progetti
calligraplan	V:0, I:2	18517	KDE	gestione progetti
gnucash	V:3, I:11	32304	GNOME	contabilità personale
homebank	V:0, I:2	1114	GTK	contabilità personale
lilypond	V:0, I:8	7337	-	music typesetter
kmyoney	V:0, I:2	12036	KDE	contabilità personale
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 per lo scanner
libreoffice-math	V:144, I:419	1963	LO	editor di equazioni matematiche e formule
calibre	V:8, I:35	55048	KDE	convertitore e gestione di una biblioteca per libri elettronici
fbreader	V:1, I:14	3074	GTK	Lettori di ebook
evince	V:105, I:313	956	GNOME	visualizzatore di documenti (pdf)
okular	V:43, I:115	14646	KDE	visualizzatore di documenti (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.

Tabella 7.3: List of notable GUI applications

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:

- [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 Tipi di carattere base

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.

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](#).

Suggerimento

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.

Suggerimento

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)`.

pacchetto	popcon	dimensioni	ans	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)

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

pacchetto	popcon	dimensioni	descrizione
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

Tabella 7.5: List of notable font environment and related packages

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](#)



avvertimento

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 Sezione [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 \(snap\)](#)

pacchetto	popcon	dimensione	descrizione
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

Tabella 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.

pacchetto	popcon	dimensione	protocols	descrizione
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

Tabella 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.

pacchetto	popcon	dimensione	comando	descrizione
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

Tabella 7.8: Elenco di metodi di connessione al server 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
```

Questo metodo può mostrare l'output da un client X remoto come se fosse connesso localmente attraverso un socket UNIX locale. See Sezione 6.3 for SSH/SSHD.



avvertimento

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 Appunti

For clipping text to clipboard, see Sezione 1.4.4.

For clipping graphics to clipboard, see Sezione 11.6.

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

pacchetto	popcon	dimensione del pac- chetto	target	descrizione
xsel	V:9, I:43	59	per X	command line interface to X selections (clipboard)
xclip	V:10, I:51	64	per 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	Console Linux	a daemon that captures mouse events on Linux console

Tabella 7.9: List of programs related to manipulating character clipboard

Capitolo 8

I18N e L10N

Il [supporto per le lingue native o M17N \(Multilingualization\)](#) per un software applicativo è ottenuto in 2 passi.

- L'internazionalizzazione (I18N): per rendere un software capace di gestire potenzialmente localizzazioni multiple.
- Localizzazione (L10N): per fare gestire dal software una localizzazione specifica.

Suggerimento

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 La localizzazione

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 Sezione [1.5.2](#)

8.1.1 Logica alla base dell'uso della localizzazione 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).

Anche un testo in semplice inglese può contenere caratteri non ASCII; le virgolette singole ricurve destra e sinistra per esempio non sono disponibili in 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 Tabella 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.

Suggerimento

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

Alcuni programmi usano più memoria dopo l'inclusione del supporto per l'internazionalizzazione. Questo avviene perché il loro codice è programmato per usare internamente [UTF-32\(UCS4\)](#) per supportare Unicode al fine di ottimizzare la velocità e consumano 4 byte per ogni dato di carattere ASCII, indipendentemente dalla localizzazione selezionata. Ancora una volta usando la localizzazione UTF-8 non si perde nulla.

8.1.2 La (ri)configurazione della localizzazione

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 Sezione 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 Codifica per i nomi di file

Per lo scambio di dati interpiattaforma (vedere Sezione 10.1.7), può essere necessario montare alcuni file system con codifiche particolari. Per esempio, `mount(8)`, se usato senza opzioni, assume che venga usata la codifica [CP437](#) per il file system `vfat`. È necessario fornire esplicitamente opzioni di montaggio per usare nomi di file [UTF-8](#) o [CP932](#).

Nota

Quando una chiavetta USB inseribile a caldo viene automaticamente montata in un ambiente desktop moderno come GNOME, si può fornire una informazione di montaggio di questo tipo cliccando con il tasto destro sull'icona del dispositivo sul desktop, cliccare sulla scheda "Drive", cliccare per espandere "Impostazioni" ed inserire "utf8" in "Opzioni di mount:". La prossima volta che questa chiavetta di memoria verrà montata, sarà abilitato il montaggio con UTF-8.

Nota

Se si sta facendo l'aggiornamento di un sistema o spostando dischi da un sistema non UTF-8, i nomi di file con caratteri non ASCII potranno essere codificati con codifiche usate una volta e ora deprecate, come [ISO-8859-1](#) o [eucJP](#). Cercare aiuto sugli strumenti di conversione dei testi per convertirli in [UTF-8](#). Vedere Sezione 11.1.

[Samba](#) usa in modo predefinito Unicode per i client più moderni (Windows NT, 200x, XP), ma usa [CP850](#) per client più vecchi (DOS e Windows 9x/Me). Questo comportamento predefinito per i client più vecchi può essere modificato usando "dos charset" nel file `/etc/samba/smb.conf`, per esempio usando [CP932](#) per il giapponese.

8.1.4 Messaggi localizzati e documentazione tradotta

Esistono le traduzioni di molti dei messaggi di testo e dei documenti che sono mostrati nel sistema Debian, come messaggi di errore, output standard dei programmi, menu e pagine di manuale. L'insieme di strumenti [GNU gettext\(1\)](#) è usato come strumento di backend per la maggior parte delle attività di traduzione.

`aptitude(8)` fornisce in "Task" → "Localizzazione" un ampio elenco di utili pacchetti binari che aggiungono alle applicazioni messaggi localizzati e che forniscono documentazione nella versione tradotta.

Per esempio, si possono ottenere i messaggi localizzati per le pagine man installando il pacchetto `manpages-LINGUA`. Per leggere le pagine man di `nomeprogramma` in italiano contenute in `/usr/share/man/it/`, eseguire il comando seguente.

```
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 Effetti della localizzazione

La disposizione dell'ordinamento dei caratteri con `sort(1)` è influenzata dalla lingua scelta dalla localizzazione. Le localizzazioni spagnola e inglese ordinano in modo diverso.

Il formato della data mostrato da `ls(1)` è influenzato dalla localizzazione. Il formato della data di `"LANG=C ls -l"` è differente da quello con `"LANG=en_US.UTF-8"` (vedere Sezione 9.3.4).

I caratteri di punteggiatura usati per i numeri sono diversi nelle varie localizzazioni. Per esempio, nella localizzazione inglese mille virgola uno è rappresentato come `"1,000.1"`, mentre nella localizzazione in italiano è mostrato come `"1.000,1"`. Si può vedere questa differenza nei programmi per fogli di calcolo.

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 L'input da tastiera

8.2.1 The keyboard input for Linux console and X Window

Il sistema Debian può essere configurato per funzionare con molte disposizioni di tastiera internazionali usando i pacchetti `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, Sezione [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 Il supporto per metodo di input con IBus

Since GNOME Desktop Environment recommends "ibus" via "gnome-shell", "ibus" is the good choice for input method.

L'input multilingua verso l'applicazione è elaborato così:

```
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 Un esempio per il giapponese

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:

pacchetto	popcon	dimensione	localizzazioni supportate
ibus	V:82, I:99	1584	infrastruttura per metodo di input che usa dbus
ibus-mozc	V:1, I:2	1006	giapponese
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	cinese (per zh_CN)
ibus-chewing	V:0, I:0	415	” ” (per zh_TW)
ibus-hangul	V:0, I:1	288	coreano
ibus-table	V:0, I:1	1809	motore di tabelle per IBus
ibus-table-thai	I:0	47	thailandese
ibus-unikey	V:0, I:0	318	vietnamita
ibus-m17n	V:0, I:1	187	multilingue: indico, arabo e altri

Tabella 8.1: List of IBus and its engine packages

1. Install the Japanese input tool package `ibus-mozc` (or `ibus-anthy`) with its recommended packages such as `im-config`.
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. Riesequire il login nell'account utente
5. Setup each input source by right clicking the GUI toolbar icon.
6. Passare da una all'altra delle fonti di input installate con SUPER-SPAZIO. (SUPER è solitamente il tasto Windows.)

Suggerimento

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.
- `im-config(8)` si comporta in modo diverso a seconda se è eseguito da root o no.
- `im-config(8)` abilita il miglior metodo di input sul sistema come impostazione predefinita senza alcuna azione dell'utente.

8.3 L'output sul display

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 (Capitolo 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 Caratteri dell'Asia dell'est con larghezza ambigua

Nella localizzazione dell'Asia dell'est i caratteri di disegno di riquadri, i caratteri greci e cirillici possono essere visualizzati più larghi della larghezza desiderata e causare un output su terminale non allineato (vedere [Unicode Standard Annex #11](#)).

Questo problema può essere aggirato:

- `gnome-terminal`: Preferences → Profiles → *Profile name* → Compatibility → Ambiguous-wide characters → Narrow
 - `ncurses`: impostare l'ambiente `export NCURSES_NO_UTF8_ACS=0`.
-

Capitolo 9

Suggerimenti per il sistema

In questa sezione vengono descritti suggerimenti base per configurare e gestire il sistema, per lo più dalla console.

9.1 The console tips

There are some utility programs to help your console activities.

pacchetto	popcon	dimensione	descrizione
mc	V:55, I:234	1492	See Sezione 1.3
bsdutils	V:708, I:999	394	<code>script</code> command to make typescript of terminal session
screen	V:114, I:283	1014	multiplexer per terminale con emulazione di terminale VT100/ANSI
tmux	V:35, I:143	830	alternativa per multiplexer per terminale (usare invece "Control-B")
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

Tabella 9.1: List of programs to support console activities

9.1.1 Registrare le attività della shell in modo pulito

Il semplice uso di `script(1)` (vedere Sezione 1.4.9) per registrare l'attività della shell produce un file con caratteri di controllo. Ciò può essere evitato usando `col(1)` nel modo seguente.

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

Fare tutto quello che si vuole ...e poi premere `Ctrl-D` per uscire da `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 Sezione 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 Il programma screen

`screen(1)` non permette solamente il funzionamento di una finestra di terminale con processi multipli, ma permette anche ai **processi in shell remote di sopravvivere a connessioni interrotte**. Quello che segue è un tipico scenario di uso di `screen(1)`.

1. Si fa il login in una macchina remota.
2. Si avvia `screen` in una console singola.
3. Si eseguono svariati programmi in finestre `screen` create con `^A c` ("Control-A" seguito da "c").
4. Ci si sposta tra le svariate finestre `screen` create con `^A n` ("Control-A" seguito da "n").
5. All'improvviso si ha la necessità di lasciare il terminale, ma non si vuole perdere il lavoro attivo e si vuole mantenere la connessione.
6. Si può **scollegare** la sessione `screen` in uno qualsiasi dei metodi seguenti.
 - Scollegare brutalmente la connessione di rete
 - Digitare `^A d` ("Control-A" seguito da "d") e fare manualmente il log out dalla connessione remota.
 - Digitare `^A DD` ("Control-A" seguito da "DD") per far sì che `screen` si scolleghi e faccia il log out.
7. Ci si ricollega alla stessa macchina remota (anche da un terminale diverso).
8. Si avvia `screen` con `screen -r`.
9. `screen` magicamente **ricollega** tutte le finestre `screen` precedente con tutti i programmi in esecuzione attivi.

Suggerimento

Con `screen` si può risparmiare sui costi di connessione per connessioni a tempo, come dial-up o conteggiate a pacchetti, perché si può lasciare un processo attivo mentre si è disconnessi e poi ricollegarsi successivamente quando è possibile connettersi di nuovo.

In una sessione `screen` tutto l'input da tastiera viene inviato alla finestra attuale, tranne per le combinazioni di tasti per i comandi. Tutte le combinazioni di tasti per i comandi di `screen` vengono inserite digitando `^A` ("Control-A") più un singolo tasto [più eventuali parametri]. Ecco alcune combinazioni di tasti importanti da ricordare.

Vedere `screen(1)` per i dettagli.

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

azione associata	significato
<code>^A ?</code>	mostra una schermata di aiuto (visualizza le associazioni di tasti)
<code>^A c</code>	crea una nuova finestra e si sposta in essa
<code>^A n</code>	va alla finestra successiva
<code>^A p</code>	va alla finestra precedente
<code>^A 0</code>	va alla finestra numero 0
<code>^A 1</code>	va alla finestra numero 1
<code>^A w</code>	mostra l'elenco delle finestre
<code>^A a</code>	invia Ctrl-A alla finestra attuale come input da tastiera
<code>^A h</code>	scrive una copia della schermata della finestra attuale in un file
<code>^A H</code>	inizia/termina la registrazione dell'attività della finestra attuale in un file
<code>^A ^X</code>	blocca il terminale (protetto da password)
<code>^A d</code>	scollega la sessione screen dal terminale
<code>^A DD</code>	scollega la sessione di screen e fa il log out

Tabella 9.2: Elenco di associazioni di tasti per screen

9.1.3 Navigating around directories

In Sezione 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`.

```
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 Personalizzare vim

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

**Attenzione**

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".

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

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 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

Tabella 9.3: Information on the initialization of vim

9.3 Registrazione e presentazione di dati

9.3.1 Demoni di registro

Molti programmi registrano le proprie attività nella directory "/var/log/".

Vedere Sezione 3.4 e Sezione 3.3.

9.3.2 Analizzatori di registro

Quelli che seguono sono alcuni analizzatori di registro degni di nota ("~Gsecurity::log-analyzer" in aptitude(8)).

Nota

[CRM114](#) fornisce un'infrastruttura basata su un linguaggio per scrivere filtri **fuzzy** con la [libreria per espressioni regolari TRE](#). Il suo utilizzo più comune è come filtro per la posta spazzatura, ma può anche essere usato come analizzatore di registro.

9.3.3 Visualizzazione personalizzata di dati di testo

Sebbene gli strumenti di paginazione, come `more(1)` e `less(1)` (vedere Sezione 1.4.5) e gli strumenti personalizzati per l'evidenziazione e la formattazione (vedere Sezione 11.1.8) possano mostrare il testo in un modo piacevole, gli editor generici (vedere Sezione 1.4.6) sono più versatili e personalizzabili.

Suggerimento

Per `vim(1)` e la sua modalità per paginatore, ossia `view(1)`, ":set hls" abilita la ricerca con evidenziazione.

²[vim-pathogen](#) was popular.

pacchetto	popcon	dimensione	descrizione
logwatch	V:16, I:18	2265	analizzatore di registro scritto in Perl con un bell'output
fail2ban	V:116, I:129	2092	strumento per interdire IP che causano errori di autenticazione multipli
analog	V:3, I:105	3534	analizzatore del registro di server web
awstats	V:9, I:14	6910	analizzatore potente e ricco di funzionalità per registro di server web
sarg	V:2, I:3	843	generatore di rapporti sull'analisi di squid
pflogsumm	V:2, I:4	111	strumento per riassunti delle voci di registro di Postfix
syslog-summary	V:0, I:2	30	riassume i contenuti di un file di registro di sistema
fwlogwatch	V:0, I:0	478	analizzatore del registro del firewall
squidview	V:0, I:1	189	monitora e analizza i file access.log di squid
swatch	V:0, I:0	101	visualizzatore di file di registro con corrispondenze con espressioni regolari, evidenziazione ed eventi
crm114	V:0, I:0	1119	Controllable Regex Mutilator e filtro per spam (CRM114)
icmpinfo	V:0, I:0	44	interpreta i messaggi ICMP

Tabella 9.4: Elenco di analizzatori del registro di sistema

9.3.4 Visualizzazione personalizzata di date e orari

The default display format of time and date by the `ls -l` command depends on the **locale** (see Sezione 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 si desidera veramente personalizzare questo formato di visualizzazione delle date e degli orari, oltre a ciò che è fatto con la **localizzazione**, si deve impostare il **valore dello stile degli orari** con l'opzione `--time-style` o con il valore di `$TIME_STYLE` (vedere `ls(1)`, `date(1)`, `info coreutils 'ls invocation'`).

valore dello stile per gli orari	localizzazione	visualizzazione di data e ora
iso	qualsiasi	01-19 00:15
long-iso	qualsiasi	2009-01-19 00:15
full-iso	qualsiasi	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	qualsiasi	19.01.09 00:15
+%d.%b.%y %H:%M	C o en_US.UTF-8	19. Jan.09 00:15
+%d.%b.%y %H:%M	es_ES.UTF-8	19. ene.09 00:15

Tabella 9.5: Display examples of time and date for the `ls -l` command with the **time style value**

Suggerimento

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

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

9.3.5 Output colorato per la shell

L'output inviato a schermo dalla shell nella maggior parte dei terminali moderni può essere colorato usando [codici di escape ANSI](#) (vedere `/usr/share/doc/xterm/ctlseqs.txt.gz`).

Per esempio, provare a fare quanto segue.

```
$ RED=$(printf "\x1b[31m")
$ NORMAL=$(printf "\x1b[0m")
$ REVERSE=$(printf "\x1b[7m")
$ echo "${RED}RED-TEXT${NORMAL} ${REVERSE}REVERSE-TEXT${NORMAL}"
```

9.3.6 Comandi colorati

I comandi colorati sono comodi per ispezionare il loro output in modo interattivo. Nel mio file `~/ .bashrc` io includo quanto segue.

```
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
```

Questo uso degli alias limita gli effetti colorati all'uso interattivo dei comandi. Ha il vantaggio, rispetto all'esportazione della variabile d'ambiente `export GREP_OPTIONS='--color=auto'`, che i colori possono essere visti in programmi di paginazione come `less(1)`. Se si desidera eliminare i colori quando si invia l'output con una pipe ad altri comandi, usare nell'esempio precedente per `~/ .bashrc` `--color=auto`.

Suggerimento

Questi alias di colorazione possono essere disattivati nell'ambiente interattivo invocando la shell con il comando `"TERM=dumb bash"`.

9.3.7 Registrare le attività dell'editor per ripetizioni complesse

È possibile registrare le attività dell'editor per ripetere azioni complesse.

Per [vim](#) fare nel modo seguente.

- `"qa"`: avviare la registrazione dei caratteri digitati in un registro chiamato `"a"`.
- `...`attività nell'editor
- `"q"`: terminare la registrazione dei caratteri digitati.
- `"@a"`: eseguire il contenuto del registro `"a"`.

Per [Emacs](#) fare nel modo seguente.

- `"C-x ("`: iniziare la definizione di una macro da tastiera.
 - `...`attività nell'editor
 - `"C-x)"`: terminare la definizione di una macro da tastiera.
 - `"C-x e"`: eseguire la definizione di una macro da tastiera.
-

9.3.8 Registrare l'immagine grafica di un'applicazione X

Ci sono alcuni modi per registrare l'immagine grafica di un'applicazione X, incluso il display di un `xterm`.

pacchetto	popcon	dimensione	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	per X
imagemagick	I:373	220	per X
scrot	V:7, I:76	123	per X

Tabella 9.6: Elenco di strumenti di manipolazione di immagini grafiche

9.3.9 Registrare i cambiamenti a file di configurazione

There are specialized tools to record changes in configuration files with help of DVCS and to make system snapshots on [Btrfs](#).

pacchetto	popcon	dimensione	descrizione
etckeeper	V:28, I:32	176	archivia i file di configurazione ed i loro metadati con Git (predefinito), Mercurial o Bazaar (nuovo)
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

Tabella 9.7: List of packages which can record configuration history

You may also think about local script Sezione [10.2.2](#) approach.

9.4 Monitorare, controllare ed avviare l'attività dei programmi

Le attività dei programmi possono essere monitorare e controllate usando strumenti specializzati.

Suggerimento

Il pacchetto `procps` fornisce strumenti estremamente di base per monitorare, controllare ed avviare le attività dei programmi. È consigliabile imparare ad usarli tutti.

9.4.1 Cronometrare un processo

Mostrare il tempo usato dal processo invocato da un 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
```

pacchetto	popcon	dimensione	descrizione
coreutils	V:901, I:999	17478	nice(1) : esegue un programma con priorità di schedulazione modificata
bsdutils	V:708, I:999	394	renice(1) : modifica la priorità di schedulazione di un processo in esecuzione
procps	V:742, I:999	1648	utilità per il file system <code>"/proc"</code> : ps(1) , top(1) , kill(1) , watch(1) , ...
psmisc	V:433, I:831	793	utilità per il file system <code>"/proc"</code> : killall(1) , fuser(1) , peekfd(1) , pstree(1)
time	V:13, I:249	129	time(1) : esegue un programma per riportare l'uso delle risorse di sistema in funzione del tempo
sysstat	V:168, I:190	1923	sar(1) , iostat(1) , mpstat(1) , ...: strumenti per le prestazioni di sistema per Linux
isag	V:0, I:4	117	Interactive System Activity Grapher (tracciamento interattivo dell'attività) per sysstat
lsof	V:393, I:944	451	lsof(8) : elenca, usando l'opzione <code>"-p"</code> , i file aperti da un processo in esecuzione
strace	V:16, I:151	2367	strace(1) : traccia le chiamate e i segnali di sistema
ltrace	V:1, I:22	363	ltrace(1) : traccia le chiamate di libreria
xtrace	V:0, I:0	353	xtrace(1) : traccia la comunicazione tra client e server X11
powertop	V:8, I:204	662	powertop(1) : informazioni sull'uso dell'alimentazione da parte del sistema
cron	V:800, I:997	263	esegue processi dal demone <code>CRON(8)</code> sullo sfondo in base ad una pianificazione
anacron	V:399, I:463	104	pianificatore di comandi in stile cron, per i sistemi che non sono in esecuzione 24 ore al giorno
at	V:154, I:278	161	at(1) o batch(1) : esegue un compito ad un orario specificato o quando il carico di sistema scende sotto un certo livello

Tabella 9.8: Elenco di strumenti per monitorare e controllare l'attività dei programmi.

valore di nice	priorità di schedulazione
19	processo dalla priorità più bassa possibile (nice - gentile)
0	processo a priorità molto alta per l'utente
-20	processo a priorità molto alta per root (not-nice - non gentile)

Tabella 9.9: Elenco di valori di nice per la priorità di schedulazione

9.4.2 Priorità di schedulazione

Per controllare la priorità di schedulazione di un processo è usato il valore di nice.

```
# nice -19 top # very nice
# nice --20 wodim -v -eject speed=2 dev=0,0 disk.img # very fast
```

A volte un valore molto alto di nice fa più male che bene al sistema; usare quindi questo comando con molta cautela.

9.4.3 Il comando ps

Il comando `ps(1)` in un sistema Debian supporta sia le funzionalità BSD sia quelle SystemV ed aiuta ad identificare l'attività dei processi in modo statico.

stile	comando tipico	funzione
BSD	<code>ps aux</code>	mostra %CPU %MEM
System V	<code>ps -efH</code>	mostra PPID

Tabella 9.10: Elenco degli stili per il comando ps

È possibile uccidere i processi figli zombie (defunti) tramite l'ID di processo del genitore identificato dal campo "PPID".

Il comando `pstree(1)` mostra un albero dei processi.

9.4.4 Il comando top

`top(1)` in sistemi Debian ha molte funzionalità e aiuta ad identificare in modo dinamico i processi che si stanno comportando in modo strano.

È un programma interattivo a tutto schermo. Si possono ottenere le informazioni di aiuto sul suo uso premendo il tasto «h» e uscire premendo il tasto «q».

9.4.5 Elencare i file aperti da un processo

È possibile elencare tutti i file aperti da un processo attraverso il suo PID (Identificativo di processo), ad esempio 1, con il comando seguente.

```
$ sudo lsof -p 1
```

Di solito il processo con PID=1 è il programma `init`.

9.4.6 Tenere traccia delle attività di un programma

Si può tenere traccia dell'attività di un programma con `strace(1)`, `ltrace(1)` o `xtrace(1)` rispettivamente per quello che riguarda chiamate e segnali di sistema, chiamate di libreria o comunicazioni tra client e server X11.

Si può tenere traccia delle chiamate di sistema del comando `ls` nel modo seguente.

```
$ sudo strace ls
```

Suggerimento

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

9.4.7 Identificazione di processi in base a file o socket

Usando `fuser(1)` è anche possibile identificare i processi in base ai file usando, ad esempio per `"/var/log/mail.log"` con il comando seguente.

```
$ sudo fuser -v /var/log/mail.log
                USER          PID ACCESS COMMAND
/var/log/mail.log:  root          2946 F.... rsyslogd
```

Come si vede il file `"/var/log/mail.log"` è aperto in scrittura dal comando `rsyslogd(8)`.

Usando `fuser(1)` si può anche identificare i processi in base ai socket, ad esempio per `"smtp/tcp"` con il comando seguente.

```
$ sudo fuser -v smtp/tcp
                USER          PID ACCESS COMMAND
smtp/tcp:       Debian-exim  3379 F.... exim4
```

Ora si può vedere che sul sistema è in esecuzione `exim4(8)` per gestire le connessioni [TCP](#) alla porta [SMTP](#) (25).

9.4.8 Ripetere un comando ad intervalli costanti

`watch(1)` esegue un programma in modo ripetitivo ad intervalli regolari mostrando il suo output sullo schermo.

```
$ watch w
```

Questo comando mostra chi è attualmente connesso al sistema in modo aggiornato ogni 2 secondi.

9.4.9 Ripetere un comando su diversi file

Ci sono svariati modi di ripetere uno stesso comando su diversi file che rispondono ad una qualche condizione, ad esempio che corrispondono al modello glob `"*.ext"`.

- Metodo del ciclo `for` nella shell (vedere Sezione [12.1.4](#)):

```
for x in *.ext; do if [ -f "$x" ]; then command "$x" ; fi; done
```

- Combinazione di `find(1)` e `xargs(1)`:

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | xargs -0 -n 1 command
```

- `find(1)` con l'opzione `"-exec"` con un comando:

```
find . -type f -maxdepth 1 -name '*.ext' -exec command '{} ' \;
```

- `find(1)` con l'opzione `"-exec"` con un breve script di shell:

```
find . -type f -maxdepth 1 -name '*.ext' -exec sh -c "command '{} ' && echo 'successful'" \;
```

Gli esempi precedenti sono stati scritti per assicurare la gestione appropriata di nomi di file particolari come quelli contenenti spazi. Per usi più avanzati di `find(1)`, vedere Sezione [10.1.5](#).

9.4.10 Avviare un programma dalla GUI

Per l'[interfaccia a riga di comando \(CLI\)](#), viene eseguito il primo programma con un nome corrispondente trovato nelle directory specificate nella variabile d'ambiente \$PATH. Vedere Sezione [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".

Per esempio, il file chromium.desktop definisce gli attributi per il «Browser Web Chromium» come «Name» per il nome di programma, «Exec» per il percorso e gli argomenti di esecuzione del programma, «Icon» per l'icona usata, ecc. (vedere la [Specificazione per le voci per desktop](#)), nel modo seguente:

```
[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
```

Questa è una descrizione molto semplificata. I file *.desktop vengono analizzati nel modo seguente.

L'ambiente desktop imposta le variabili d'ambiente \$XDG_DATA_HOME e \$XDG_DATA_DIR. Per esempio, in GNOME 3:

- \$XDG_DATA_HOME non è impostata. (Viene usato il valore predefinito: \$HOME/.local/share.)
- \$XDG_DATA_DIRS è impostata a /usr/share/gnome:/usr/local/share:/usr/share/.

Perciò le directory base (vedere la [XDG Base Directory Specification](#)) e le directory applications sono le seguenti.

- \$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/

I file *.desktop vengono analizzati all'interno di queste directory applications in tale ordine.

Suggerimento

Una voce del menu GUI personalizzata dell'utente può essere creata aggiungendo un file *.desktop nella directory \$HOME/.local/share/applications/.

Suggerimento

In modo analogo, se viene creato un file *.desktop nella directory autostart all'interno di queste directory di base, il programma specificato nel file *.desktop viene eseguito automaticamente all'avvio dell'ambiente desktop. Vedere la specifica [Desktop Application Autostart Specification](#).

Suggerimento

In modo simile, se viene creato un file `*.desktop` nella directory `$HOME/Desktop` e l'ambiente desktop è configurato per supportare la funzionalità di avviatore delle icone, il programma specificato in tale file viene eseguito quando si fa clic sull'icona. Notare che il nome effettivo della directory `$HOME/Desktop` dipende dalla localizzazione. Vedere `xdg-user-dirs-update(1)`.

9.4.11 Personalizzare i programmi da avviare

Alcuni programmi avviano automaticamente altri programmi. Quelli che seguono sono alcuni punti fondamentali per la personalizzazione di questo processo.

- Menu di configurazione delle applicazioni:
 - Desktop GNOME3: "Impostazioni di sistema" → "System" → "Dettagli" → "Applicazioni predefinite"
 - Desktop KDE: "K" → "Centro di controllo" → "Componenti di KDE" → "Scelta componenti"
 - Browser Iceweasel: "Modifica" → "Preferenze" → "Applicazioni"
 - mc(1): `"/etc/mc/mc.ext"`
- Variabili d'ambiente quali `"$BROWSER"`, `"$EDITOR"`, `"$VISUAL"` e `"$PAGER"` (vedere `environ(7)`)
- Il sistema `update-alternatives(1)` per i programmi come `"editor"`, `"view"`, `"x-www-browser"`, `"gnome-www-browser"` e `"www-browser"` (vedere Sezione 1.4.7)
- Il contenuto dei file `"~/ .mailcap"` e `"/etc/mailcap"` che associano i tipi MIME con programmi (vedere `mailcap(5)`)
- Il contenuto dei file `"~/ .mime.types"` e `"/etc/mime.types"` che associano le estensioni dei nomi di file con tipi MIME (vedere `run-mailcap(1)`)

Suggerimento

`update-mime(8)` aggiorna il file `"/etc/mailcap"` usando il file `"/etc/mailcap.order"` (vedere `mailcap.order(5)`).

Suggerimento

Il pacchetto `debianutils` fornisce `sensible-browser(1)`, `sensible-editor(1)` e `sensible-pager(1)` che prendono decisioni sensate riguardo, rispettivamente, a quale browser web, editor e paginatore invocare. La lettura di questi script di shell è raccomandata.

Suggerimento

Per eseguire un'applicazione per console, come `mutt`, come applicazione preferita in X si dovrebbe creare un'applicazione X nel modo seguente ed impostare `"/usr/local/bin/mutt-term"` come applicazione preferite da avviare come descritto in precedenza.

```
# cat /usr/local/bin/mutt-term <<EOF
#!/bin/sh
gnome-terminal -e "mutt @$@"
EOF
chmod 755 /usr/local/bin/mutt-term
```

valore del segnale	nome del segnale	azione	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 <code>dump core</code>	quit from keyboard (CTRL - \)
9	<code>SIGKILL</code>	terminate the process	unblockable kill signal
15	<code>SIGTERM</code>	terminate the process	blockable termination signal

Tabella 9.11: Elenco dei segnali usati comunemente con il comando kill

9.4.12 Uccidere un processo

Per uccidere un processo (o inviare ad esso un segnale) in base al suo ID (identificativo) usare `kill(1)`.

Per fare la stessa cosa ma in base al nome del comando del processo o ad altri attributi, usare `killall(1)` o `pkill(1)`.

9.4.13 Pianificare compiti da eseguire una volta sola

Per pianificare un compito da eseguire una volta soltanto eseguire il comando `at(1)` nel modo seguente.

```
$ echo 'command -args' | at 3:40 monday
```

9.4.14 Pianificare compiti in modo regolare

Per pianificare compiti in modo regolare usare `cron(8)`. Vedere `crontab(1)` e `crontab(5)`.

Si può pianificare l'esecuzione di processi come utente normale, ad esempio l'utente `pippo`, creando un file `crontab(5)` come `"/var/spool/cron/crontabs/pippo"` con il comando `"crontab -e"`.

Quello seguente è un esempio di file `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
```

Suggerimento

Per i sistemi non in esecuzione in maniera continuata, installare il pacchetto `anacron` per pianificare l'esecuzione di comandi periodici, in maniera il più possibile vicina agli intervalli specificati, in base a quanto permesso dal tempo di attività della macchina. Vedere `anacron(8)` e `anacrontab(5)`.

Suggerimento

Gli script con compiti pianificati di amministrazione del sistema possono essere eseguiti periodicamente dall'account di `root`, ponendoli in `"/etc/cron.hourly/"`, `"/etc/cron.daily/"`, `"/etc/cron.weekly/"` o `"/etc/cron.monthly/"`. L'orario di esecuzione di questi script può essere personalizzato con `"/etc/crontab"` e `"/etc/anacrontab"`.

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

tasto premuto dopo Alt-R_Sist	descrizione dell'azione
k	kill all processes on the current virtual console (SAK)
s	sincronizza tutti i file system montati per evitare corruzione di dati
u	monta nuovamente tutti i file system in sola lettura (u mount)
r	ripristina la tastiera dalla modalità raw dopo un crash di X

Tabella 9.12: List of notable SAK command keys

9.4.15 Tasto Alt-SysRq

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

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

Suggerimento

Si può usare la funzione Alt-R_Sist da un terminale SSH, ecc. scrivendo su `"/proc/sysrq-trigger"`. Per esempio, `"echo s > /proc/sysrq-trigger; echo u > /proc/sysrq-trigger"` dal prompt di shell di root sincronizza ed esegue `umount` per tutti i file system montati.

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 Suggerimenti per l'amministrazione del sistema

9.5.1 Chi è nel sistema?

Si può controllare chi è connesso al sistema nei modi seguenti.

- `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 l'elenco degli utenti che hanno tentato il login non riuscendovi più recentemente.

Suggerimento

Queste informazioni sugli utenti sono contenute in `"/var/run/utmp"` e `"/var/log/wtmp"`. Vedere `login(1)` e `utmp(5)`.

9.5.2 Avvertire tutti gli utenti

Si può mandare un messaggio a tutti gli utenti che sono connessi al sistema con `wall(1)` nel modo seguente.

```
$ echo "We are shutting down in 1 hour" | wall
```

9.5.3 Identificazione dell'hardware

Per i dispositivi di tipo simil-PCI ([AGP](#), [PCI-Express](#), [CardBus](#), [ExpressCard](#), ecc.), `lspci(8)` (probabilmente con l'opzione "-nn") è un buon punto di partenza per l'identificazione dell'hardware.

In alternativa, si può identificare l'hardware leggendo il contenuto di `"/proc/bus/pci/devices"` o sfogliando l'albero di directory in `"/sys/bus/pci"` (vedere Sezione [1.2.12](#)).

pacchetto	popcon	dimensione	descrizione
pciutils	V:200, I:991	196	utilità PCI Linux: <code>lspci(8)</code>
usbutils	V:73, I:858	325	utilità Linux USB: <code>lsusb(8)</code>
nvme-cli	V:2, I:10	779	NVMe utilities for Linux: <code>nvme(1)</code>
pcmciautils	V:11, I:17	97	utilità PCMCIA per Linux: <code>pccardctl(8)</code>
scsitools	V:0, I:3	375	raccosta di strumenti per la gestione dell'hardware SCSI: <code>lsscsi(8)</code>
procinfo	V:0, I:12	135	informazioni sul sistema ottenute da <code>"/proc"</code> : <code>lsdev(8)</code>
lshw	V:11, I:96	850	informazioni sulla configurazione hardware: <code>lshw(1)</code>
discover	V:37, I:948	90	sistema di identificazione dell'hardware: <code>discover(8)</code>

Tabella 9.13: Elenco di strumenti per l'identificazione dell'hardware

9.5.4 Configurazione dell'hardware

Sebbene nei moderni sistemi desktop con interfaccia grafica, come GNOME e KDE, la maggior parte della configurazione dell'hardware possa essere gestita attraverso strumenti di configurazione con interfaccia grafica da essi forniti, è bene conoscere alcuni metodi di base di configurazione.

pacchetto	popcon	dimensione	descrizione
console-setup	V:133, I:960	419	utilità Linux per i tipi di carattere e la mappa di tastiera in console
x11-xserver-utils	V:272, I:516	511	utilità del server X: <code>xset(1)</code> , <code>xmodmap(1)</code>
acpid	V:135, I:281	176	demone per gestire eventi inviati dall'ACPI (Advanced Configuration and Power Interface, interfaccia avanzata per configurazione ed energia)
acpi	V:15, I:266	45	utilità per mostrare informazioni sui device ACPI
sleepd	V:0, I:0	86	demone per sospendere un portatile durante l'inattività
hdparm	V:446, I:717	256	ottimizzazione degli accessi al disco fisso (vedere Sezione 9.6.9)
smartmontools	V:150, I:208	2117	controllo e monitoraggio dei sistemi di archiviazione usando S.M.A.R.T.
setserial	V:5, I:9	117	raccolta di strumenti per la gestione delle porte seriali
memtest86+	V:1, I:28	2391	raccolta di strumenti per la gestione di hardware di memoria
scsitools	V:0, I:3	375	raccolta di strumenti per la gestione di hardware SCSI
setcd	V:0, I:1	35	ottimizzazione dell'accesso alle unità CD
big-cursor	I:1	27	puntatori del mouse più grandi per X

Tabella 9.14: Elenco di strumenti di configurazione dell'hardware

[ACPI](#) è un'infrastruttura per il sistema di gestione dell'energia più nuovo di [APM](#)

Suggerimento

La modifica della frequenza della CPU nei sistemi moderni è controllata da moduli del kernel come `acpi_cpufreq`.

9.5.5 Orario di sistema e hardware

I comandi seguenti impostano l'ora di sistema e hardware a MM/GG hh:mm, AAAA.

```
# date MMDDhhmmCCYY
# hwclock --utc --systohc
# hwclock --show
```

Nei sistemi Debian, gli orari sono mostrati normalmente come ora locale, ma l'ora di sistema e quella hardware usano di solito l'ora [UTC\(GMT\)](#).

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

Il comando seguente riconfigura il fuso orario utilizzato dal sistema Debian.

```
# dpkg-reconfigure tzdata
```

Se si desidera tenere aggiornata l'ora di sistema attraverso la rete, si consideri l'uso del servizio [NTP](#) con pacchetti come `ntp`, `ntpdate` e `chrony`.

Suggerimento

In [systemd](#), usare invece `systemd-timesyncd` per l'ora di rete. Vedere `systemd-timesyncd(8)`.

Vedere la documentazione seguente.

- [Managing Accurate Date and Time HOWTO](#)
- [Progetto servizi NTP pubblici](#)
- Il pacchetto `ntp-doc`

Suggerimento

`ntptrace(8)`, nel pacchetto `ntp` può tracciare una catena di server NTP all'indietro fino alla fonte originaria.

9.5.6 Configurazione del terminale

Ci sono diversi componenti per configurare le funzionalità della console a caratteri e il sistema `ncurses(3)`.

- Il file `/etc/terminfo/*/*` (`terminfo(5)`)
- La variabile d'ambiente `$TERM` (`term(7)`)
- `setterm(1)`, `stty(1)`, `tic(1)` e `toe(1)`

:(Se, con un `xterm` non Debian, la voce `terminfo` per `xterm` non funziona, cambiare il tipo di terminale `$TERM` da `xterm` ad una delle versioni con funzionalità limitate come `xterm-r6` quando si fa il login ad un sistema Debian da remoto. Per ulteriori vedere `/usr/share/doc/libncurses5/FAQ`).

9.5.7 L'infrastruttura audio

I driver di dispositivo per le schede audio per l'attuale Linux sono forniti da [ALSA \(Advanced Linux Sound Architecture\)](#). ALSA fornisce una modalità di emulazione per la compatibilità con il precedente sistema [OSS \(Open Sound System\)](#).

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.

Normalmente esiste un motore audio comune per ciascun ambiente desktop. Ciascun motore audio usato dalle applicazioni può scegliere di connettersi a diversi server audio.

Suggerimento

Per testare l'altoparlante usare `cat /dev/urandom > /dev/audio` oppure `speaker-test(1)` (^C per interrompere).

Suggerimento

Se non si ottiene l'audio, è possibile che l'altoparlante sia connesso ad un output impostato come muto. I moderni sistemi sonori hanno svariati output. `alsamixer(1)` nel pacchetto `alsa-utils` è utile per configurare le impostazioni del volume e di muto.

pacchetto	popcon	dimensione	descrizione
alsa-utils	V:337, I:456	2283	utilità per configurare ed usare ALSA
oss-compat	V:1, I:27	20	compatibilità OSS in ALSA per prevenire errori <code>"/dev/dsp not found"</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	server JACK (JACK Audio Connection Kit) (bassa latenza)
libjack0	V:1, I:12	338	libreria JACK (JACK Audio Connection Kit) (bassa latenza)
libgststreamer1.0-0	V:374, I:567	5283	GStreamer : motore audio di GNOME
libphonon4	I:101	680	Phonon : motore audio di KDE

Tabella 9.15: Elenco di pacchetti relativi all'audio

9.5.8 Disabilitare il salvaschermo

Per disabilitare il salvaschermo usare i comandi seguenti.

9.5.9 Disabilitare i bip sonori

Per disabilitare i bip sonori è sempre possibile disconnettere l'altoparlante del PC; la rimozione del modulo `pcspkr` del kernel fa stessa cosa per conto dell'utente.

Il comando seguente evita che il programma `readline(3)` usato da `bash(1)` emetta suoni bip quando incontra un carattere di allerta (ASCII=7).

```
$ echo "set bell-style none">> ~/.inputrc
```

ambiente	comando
Console Linux	setterm -powersave off
X Window (disabilitare il salvaschermo)	xset s off
X Window (disabilitare DPMS)	xset -dpms
X Window (configurazione tramite GUI del salvaschermo)	xscreensaver-command -prefs

Tabella 9.16: Elenco di comandi per disabilitare il salvaschermo

9.5.10 Uso della memoria

Ci sono 2 risorse disponibili per l'utente per ottenere la situazione dell'uso della memoria.

- I messaggi di avvio del kernel nel file `/var/log/dmesg` contengono la dimensione esatta della memoria disponibile.
- `free(1)` e `top(1)` mostrano informazioni sulle risorse di memoria nel sistema mentre è in funzione.

Ecco un esempio.

```
# 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
```

Ci si potrebbe chiedere perché `dmesg` dice che ci sono 990 MB liberi e `free -k` dice che sono liberi 320 MB. Ne mancano più di 600 MB.

Non ci si deve preoccupare della grande dimensione del valore "used" (usata) e del piccolo valore di "free" (libera) nella riga "Mem:", ma si può invece leggere la riga sottostante (con 675404 e 321780 nell'esempio precedente) e rilassarsi.

Per il mio MacBook con 1GB=1048576k DRAM (il sistema video ne ruba un po'), vedo le informazioni seguenti.

fonte	dimensione
Dimensione totale in <code>dmesg</code>	1016784k = 1GB - 31792k
Libera in <code>dmesg</code>	990528k
Dimensione totale nella shell	997184k
Libera nella shell	20256k (ma in effetti 321780k)

Tabella 9.17: Elenco di dimensioni della memoria riportate

9.5.11 Verifica della sicurezza e dell'integrità del sistema

Una cattiva manutenzione del sistema può esporlo ad attacchi esterni.

Per verificare la sicurezza e l'integrità del sistema, si dovrebbe iniziare dai punti seguenti.

- Il pacchetto `debsums`; vedere `debsums(1)` e Sezione [2.5.2](#).
- Il pacchetto `chkrootkit`; vedere `chkrootkit(1)`.
- La famiglia di pacchetti `clamav`; vedere `clamscan(1)` e `freshclam(1)`.
- [FAQ Debian Security](#).

pacchetto	popcon	dimensione	descrizione
logcheck	V:8, I:10	102	demone per inviare via posta anomalie nei file di registro di sistema all'amministratore
debsums	V:5, I:44	108	utilità per controllare i file dei pacchetti installati con le somme di controllo MD5
chkrootkit	V:5, I:24	971	rilevatore di rootkit
clamav	V:12, I:57	775	utilità anti-virus per Unix - interfaccia a riga di comando
tiger	V:2, I:3	7822	riporta vulnerabilità nella sicurezza di sistema
tripwire	V:2, I:3	11618	strumento di controllo dell'integrità di file e directory
john	V:2, I:12	460	strumento di violazione delle password attive
aide	V:1, I:2	2049	AIDE - ambiente avanzato di rilevamento delle intrusioni - binario statico
integrit	V:0, I:0	329	programma di controllo dell'integrità di file
crack	V:0, I:1	152	programma per indovinare password

Tabella 9.18: Elenco di strumenti per verificare la sicurezza e l'integrità del sistema

- [Manuale Securing Debian](#).

Con il piccolo script seguente è possibile controllare la presenza di tipici errori con permessi di scrittura per tutti per i file sbagliati.

```
# find / -perm 777 -a \! -type s -a \! -type l -a \! \! \! -type d -a -perm 1777 \)
```



Attenzione

Data che il pacchetto `debsums` usa le somme di controllo [MD5](#) salvate localmente, non può essere completamente affidabile come strumento di controllo della sicurezza del sistema contro attacchi malevoli.

9.6 Suggerimenti per l'archiviazione dei dati

L'avvio del sistema con un [CD live](#) Linux o un [CD del debian-installer](#) in modalità ripristino rende semplice la riconfigurazione dell'archiviazione dei dati sul dispositivo di avvio.

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 Uso dello spazio su disco

L'uso dello spazio su disco può essere valutato con i programmi forniti dai pacchetti `mount`, `coreutils` e `xdu`:

- `mount(8)` mostra tutti i file system (= dischi) montati.
- `df(1)` mostra l'uso dello spazio su disco da parte dei file system.
- `du(1)` mostra l'uso dello spazio su disco da parte dell'albero di directory.

Suggerimento

Si può indirizzare l'output di `du(8)` a `xdu(1x)` per far sì che produca la sua rappresentazione grafica e interattiva, usando `"du -k . |xdu"`, `"sudo du -k -x / |xdu"`, ecc.

9.6.2 Configurazione del partizionamento dei dischi

Per la configurazione del [partizionamento dei dischi](#), benché `fdisk(8)` sia considerato lo strumento standard, `parted(8)` merita un po' di attenzione. "Dati di partizionamento del disco", "tabella delle partizioni", "mappa delle partizioni" e "etichetta del disco" sono tutti sinonimi.

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.

Sebbene `fdisk(8)` sia stato lo strumento standard per il partizionamento dei dischi, `parted(8)` lo sta sostituendo.

pacchetto	popcon	dimensione	descrizione
util-linux	V:899, I:999	4599	utilità di sistema varie inclusi <code>fdisk(8)</code> e <code>cdisk(8)</code>
parted	V:367, I:548	307	programma GNU Parted per il ridimensionamento delle partizioni dei dischi
gparted	V:16, I:127	2063	editor delle partizioni di GNOME basato su <code>libparted</code>
gdisk	V:298, I:501	880	partition editor for the GPT/MBR hybrid disk
kpartx	V:19, I:30	87	programma per creare la mappatura a device per le partizioni

Tabella 9.19: Elenco di pacchetti di gestione delle partizioni dei dischi



Attenzione

Sebbene `parted(8)` sostenga di creare e ridimensionare anche i file system, è più sicuro fare queste cose usando gli strumenti specializzati meglio mantenuti, come `mkfs(8)` (`mkfs.msdos(8)`, `mkfs.ext2(8)`, `mkfs.ext3(8)`, `mkfs.ext4(8)`, ...) e `resize2fs(8)`.

Nota

Per poter commutare tra [GPT](#) e [MBR](#), è necessario cancellare direttamente i primi pochi blocchi del contenuto del disco (vedere Sezione [9.8.6](#)) e usare `parted /dev/sdx mklabel gpt` o `parted /dev/sdx mklabel msdos`, per fare il cambiamento. Notare che in questo contesto è usato "msdos" per [MBR](#).

9.6.3 Accedere alle partizioni usando 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)` con l'opzione `"-U"` può montare un device a blocchi usando l'[UUID](#) invece di usare il suo nome di file come `"/dev/sda3"`.
- Il file `"/etc/fstab"` (vedere `fstab(5)`) può usare gli [UUID](#).
- Anche i bootloader (Sezione [3.1.2](#)) possono usare gli [UUID](#).

Suggerimento

Si può scoprire l'[UUID](#) di un device a blocchi speciale con `blkid(8)`. You can also probe it and other information with `"lsblk -f"`.

9.6.4 LVM2

LVM2 è un [gestore di volumi logici](#) per il kernel Linux. Con LVM2 si possono creare partizioni dei dischi in volumi logici invece che sugli hard disk fisici.

LVM richiede quanto segue.

- la gestione di device-mapper nel kernel Linux (predefinita per i kernel Debian)
- la libreria per la gestione di device-mapper in spazio utente (pacchetto `libdevmapper*`)
- gli strumenti LVM2 in spazio utente (pacchetto `lvm2`)

Per iniziare a comprendere LVM2 guardare le pagine di manuale seguenti.

- `lvm(8)`: Principi di base del meccanismo LVM2 (elenco di tutti i comandi di LVM2)
- `lvm.conf(5)`: File di configurazione per LVM2
- `lvs(8)`: Riporta informazioni sui volumi logici
- `vgs(8)`: Riporta informazioni sui gruppi di volumi
- `pvs(8)`: Riporta informazioni sui volumi fisici

9.6.5 Configurazione del file system

Per il file system [ext4](#), il pacchetto `e2fsprogs` fornisce gli strumenti seguenti.

- `mkfs.ext3(8)` per creare nuovi file system [ext4](#)
- `fck.ext4(8)` per controllare e riparare file system [ext4](#) esistenti
- `tune2fs(8)` per configurare i superblocchi di un file system [ext4](#)
- `debugfs(8)` per fare il debug di file system [ext4](#) in modo interattivo. (Ha un comando `undeL` per ripristinare file eliminati.)

I comandi `mkfs(8)` e `fck(8)` sono forniti dal pacchetto `e2fsprogs` come front-end per vari programmi dipendenti dal file system (`mkfs.tipofs` e `fck.tipofs`). Per il file system [ext4](#), sono `mkfs.ext4(8)` e `fck.ext4(8)` (sono un collegamento simbolico a `mke2fs(8)` e `e2fck(8)`).

Sono disponibili comandi simili per ciascun file system supportato da Linux.

Suggerimento

Il file system [Ext4](#) è il predefinito per il sistema Linux e il suo uso è caldamente raccomandato a meno che non siano specifiche ragioni per non farlo.

[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.

Alcuni strumenti permettono l'accesso a file system non supportati del kernel Linux (vedere Sezione [9.8.2](#)).

pacchetto	popcon	dimensione	descrizione
e2fsprogs	V:616, I:999	1511	utilità per i file system ext2/ext3/ext4
btrfs-progs	V:44, I:68	3959	utilità per i file system Btrfs
reiserfsprogs	V:15, I:29	1132	utilità per i file system Reiserfs
zfsutils-linux	V:19, I:23	1497	utilities for the OpenZFS filesystem
dosfstools	V:132, I:508	315	utilità per i file system 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-utils	V:19, I:430	231	utilities for the exFAT filesystem maintained by the exfat-fuse author.
xfsprogs	V:22, I:101	3281	utilità per i file system 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	utilità per i file system JFS (IBM: AIX, OS/2)
reiser4progs	V:0, I:4	1373	utilità per i file system Reiser4
hfsprogs	V:0, I:7	389	utilità per i file system HFS e HFS Plus (Apple: Mac OS)
zerofree	V:3, I:100	25	programma per impostare a 0 i blocchi liberi in file system ext2/3/4

Tabella 9.20: Elenco di pacchetti di gestione dei file system

9.6.6 Creare file system e verificarne l'integrità

In un sistema Linux, il comando `mkfs(8)` crea i file system ed il comando `fsck(8)` fornisce funzioni di controllo dell'integrità e di riparazione dei file system.

Debian ora in modo predefinito non fa `fsck` periodici dopo la creazione del file system.



Attenzione

In generale l'esecuzione di `fsck` su **file system montati** non è sicura.

Suggerimento

Si può eseguire in modo sicuro il comando `fsck(8)` su tutti i file system, incluso il file system radice, al riavvio impostando "enable_periodic_fsck" in `/etc/mke2fs.conf` e il conteggio massimo di montaggi a 0 usando `"tune2fs -c0 /dev/nome_partizione"`. Vedere `mke2fs.conf(5)` e `tune2fs(8)`.

Per vedere i risultati del comando `fsck(8)` avviato dallo script di avvio, controllare i file in `/var/log/fsck/`.

9.6.7 Ottimizzare il file system con opzioni di mount

La configurazione statica di base del file system è fornita da `«/etc/fstab»`. Ad esempio,

```
«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
```

Suggerimento

Per identificare un dispositivo a blocchi si può usare il suo **UUID** (vedere Sezione 9.6.3) invece del normale nome di device a blocchi quale `«/dev/sda1»`, `«/dev/sda2»`, ...

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

See `fstab(8)` and `mount(8)`.

9.6.8 Ottimizzare il file system tramite il superblocco

Le caratteristiche di un file system possono essere ottimizzate attraverso il suo superblocco usando il comando `tune2fs(8)`.

- L'esecuzione di `sudo tune2fs -l /dev/hda1` mostra il contenuto del superblocco del file system in `/dev/hda1`.
- L'esecuzione di `sudo tune2fs -c 50 /dev/hda1` cambia per `/dev/hda1` la frequenza dei controlli dei file system (l'esecuzione di `fsck` all'avvio) a 50 avvii.
- L'esecuzione di `sudo tune2fs -j /dev/hda1` aggiunge la funzionalità di journaling al file system in `/dev/hda1`, cioè converte il file system da `ext2` a `ext3`. (Eseguire questo comando su file system non montati.)
- L'esecuzione di `sudo tune2fs -O extents,uninit_bg,dir_index /dev/hda1 && fsck -pf /dev/hda1` converte il filesystem in `/dev/hda1` da `ext3` a `ext4`. (Eseguire questo comando su file system non montati.)

Suggerimento

Nonostante il suo nome, `tune2fs(8)` non funziona soltanto sui file system `ext2`, ma anche sui file system `ext3` e `ext4`.

9.6.9 Ottimizzare il disco rigido



avvertimento

Prima di giocare con la configurazione dei dischi controllare il proprio hardware e leggere la pagina man di `hdparm(8)`, perché è una cosa piuttosto pericolosa per l'integrità dei dati.

Si può testare la velocità di accesso ai dischi di un disco rigido, ad esempio `/dev/hda` con `hdparm -tT /dev/hda`. È possibile velocizzare alcuni dischi fissi connessi con (E)IDE con `hdparm -q -c3 -d1 -u1 -m16 /dev/hda` che abilita il "supporto (E)IDE per I/O a 32 bit", l'uso dell'opzione `using_dma`, imposta l'opzione `interrupt-unmask` e imposta l'"I/O di settori multipli a 16" (pericoloso!).

Si possono testare le capacità della cache in scrittura di un disco fisso, ad esempio `/dev/sda`, con `hdparm -W /dev/sda`. Si può disabilitare la funzionalità di cache in scrittura con `hdparm -W 0 /dev/sda`.

Potrebbe essere possibile leggere CDRom masterizzati male in unità CD-ROM moderne ad alta velocità rallentandole con `setcd -x 2`.

9.6.10 Ottimizzare le unità a stato solido

[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 Usare SMART per prevedere danni ai dischi fissi

Con il demone `smartd` (8) è possibile monitorare e registrare i dischi fissi che sono conformi a [SMART](#).

1. Abilitare la funzionalità [SMART](#) nel [BIOS](#).
 2. Installare il pacchetto `smartmontools`.
 3. Identificare i dispositivi dei dischi fissi usando `df(1)`.
 - Si supponga che uno dei dispositivi dei dischi fissi da monitorare sia `/dev/hda`.
-

4. Controllare l'output di `smartctl -a /dev/hda` per vedere se la funzionalità [SMART](#) è veramente abilitata.
 - Se non la è, abilitarla con `smartctl -s on -a /dev/hda`.
5. Abilitare l'esecuzione del demone `smartd(8)` nel modo seguente.
 - Rimuovere il carattere iniziale di commento dalla riga `start_smartd=yes` nel file `/etc/default/smartmontools`
 - restart the `smartd(8)` daemon by `sudo systemctl restart smartmontools`.

Suggerimento

Il demone `smartd(8)` può essere personalizzato con il file `/etc/smartd.conf`, incluso per ciò che riguarda le notifiche dei messaggi di avvertimento.

9.6.12 Specificare una directory per l'archiviazione di dati temporanei usando `$TMPDIR`

Le applicazioni creano file temporanei normalmente nella directory di memorizzazione temporanea `«/tmp»`. Se `«/tmp»` non contiene abbastanza spazio, si può specificare una directory di memorizzazione temporanea usando la variabile `$TMPDIR` per i programmi che si comportano in modo corretto.

9.6.13 Espandere lo spazio di archiviazione utilizzabile con LVM

Le partizioni create al momento dell'installazione su [LVM \(Logical Volume Manager\)](#) (funzionalità di Linux) possono essere facilmente ridimensionate concatenando ad esse delle estensioni o suddividendo le loro estensioni su dispositivi di archiviazione multipli senza riconfigurazioni importanti del sistema.

9.6.14 Espandere lo spazio di archiviazione utilizzabile montando un'altra partizione

Se è disponibile una nuova partizione vuota (per esempio `/dev/sdx`), la si può formattare con `mkfs.ext4(1)` e montarla con `mount(8)` in una directory in cui è necessario avere più spazio. (È necessario copiare il contenuto originale della directory.)

```
$ 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
```

Suggerimento

In alternativa si può montare un file immagine vuoto del disco (vedere Sezione [9.7.5](#)) come device loop (vedere Sezione [9.7.3](#)). Il reale uso del disco cresce mano a mano che vengono archiviati i dati.

9.6.15 Espandere lo spazio di archiviazione utilizzabile montando un collegamento ad un'altra directory

Se è disponibile una directory vuota (ad esempio `/percorso/della/dir-vuota`) in un'altra partizione con spazio disponibile, si può, usando `mount(8)` con l'opzione `--bind`, montarla in una directory (ad esempio `dir-di-lavoro`) dove è necessario più spazio.

```
$ sudo mount --bind /path/to/emp-dir work-dir
```

9.6.16 Espandere lo spazio di archiviazione utilizzabile montando un'altra directory da sovrapporre

Se è disponibile dello spazio utilizzabile in un'altra partizione (ad esempio `"/percorso/della/vuota"` e `"/percorso/di/lavoro"`) si può creare in essa una directory e impilarla in una vecchia directory (es., `"/percorso/della/vecchia"`) in cui si ha bisogno di spazio usando [OverlayFS](#) con un kernel Linux 3.18 o successivo (Debian Stretch 9.0 o successiva).

```
$ sudo mount -t overlay overlay \
  -olowerdir=/path/to/old-dir,upperdir=/path/to/empty,workdir=/path/to/work
```

Qui, `"/percorso/della/vuota"` e `"/percorso/di/lavoro"` devono essere nella partizione abilitata in lettura e scrittura da scrivere in `"/percorso/della/vecchia"`.

9.6.17 Espandere lo spazio di archiviazione utilizzabile usando collegamenti simbolici



Attenzione

Questo è un metodo deprecato. Certo software può non funzionare bene con i "collegamenti simbolici ad una directory". Usare invece gli approcci che usano il "mount" descritti in precedenza.

Se è disponibile una directory vuota (ad esempio `"/percorso/della/dir-vuota"` in un'altra partizione con spazio disponibile, si può creare un collegamento simbolico alla directory con `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
```



avvertimento

Non usare un "collegamento simbolico ad una directory" per le directory gestite dal sistema, come `"/opt"`. Un collegamento simbolico simile potrebbe essere sovrascritto quando il sistema viene aggiornato.

9.7 Immagine del disco

Questa sezione tratta della manipolazione di immagini di dischi.

9.7.1 Creare un file con un'immagine di disco

Si può creare un file di immagine del disco, `"disco.img"`, di un dispositivo non montato, ad esempio la seconda unità SCSI o serial ATA `"/dev/sdb"` usando `cp(1)` o `dd(1)` nel modo seguente.

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

Si può creare l'immagine del disco del [MBR \(master boot record\)](#) dei PC tradizionali (vedere Sezione 9.6.2), che risiede nel primo settore del disco IDE primario usando `dd(1)` nel modo seguente.

```
# 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": MBR con tabella delle partizioni
- "mbr-nopart.img": MBR senza tabella delle partizioni
- "mbr-part.img": solamente la tabella delle partizioni nell'MBR

Se il disco di avvio è un un dispositivo SCSI o serial ATA, sostituire "/dev/hda" con "/dev/sda".

Se si sta creando l'immagine di una partizione del disco originale, sostituire "/dev/hda" con "/dev/hda1", ecc.

9.7.2 Scrivere direttamente sul disco

Il file immagine del disco, "disk.img" può essere scritto in un dispositivo non montato, ad esempio la seconda unità SCSI "/dev/sdb" di dimensione corrispondente nel modo seguente.

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

Analogamente il file immagine di una partizione del disco, "partizione.img" può essere scritto in una partizione non montata, ad esempio la prima partizione della seconda unità SCSI "/dev/sdb1" di dimensione corrispondente nel modo seguente.

```
# dd if=partizione.img of=/dev/sdb1
```

9.7.3 Montare un file con un'immagine di disco

Un'immagine di disco "partizione.img" contenente l'immagine di un'unica partizione, può essere montata e smontata usando il [device loop](#) nel modo seguente.

```
# losetup -v -f partizione.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
```

Questo può essere semplificato nel modo seguente.

```
# mkdir -p /mnt/loop0
# mount -t auto -o loop partizione.img /mnt/loop0
...hack...hack...hack
# umount partizione.img
```

Ogni partizione di un'immagine di disco "disk.img" contenente più partizioni, può essere montata usando il [device loop](#). Dato che quest'ultimo non gestisce in modo predefinito le partizioni, è necessario riconfigurarli nel modo seguente.

```
# 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
```

Ora il device loop può gestire fino a 16 partizioni.

```
# 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
```

In alternativa, un risultato simile può essere ottenuto utilizzando i [device mapper](#) creati da `kpartx(8)`, contenuto nel pacchetto `kpartx`, nel modo seguente.

```
# 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

È anche possibile montare una singola partizione di una tale immagine di disco con un [device loop](#) indicando un offset per saltare l'[MBR](#), ecc., ma questo metodo è più esposto agli errori.

9.7.4 Pulire un file con un'immagine di disco

Si può pulire un file con immagine di disco "disco.img" da tutti i file cancellati creando un'immagine pulita "nuova.img" nel modo seguente.

```
# 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 "disco.img" è in ext2, ext3 o ext4, si può anche usare `zerofree(8)`, contenuto nel pacchetto `zerofree`, nel modo seguente.

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

9.7.5 Creare un file con immagine di disco vuoto

Si può creare un'immagine di disco vuota "disk.img", che può crescere fino a 5GiB, usando `dd(1)` nel modo seguente.

```
$ dd bs=1 count=0 if=/dev/zero of=disk.img seek=5G
```

È possibile creare un file system `ext4` in questa immagine di disco, "disk.img" usando il [device loop](#) nel modo seguente.

```
# 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
```

La dimensione del file "disk.img" è di 5.0GiB e il suo effettivo uso del disco è di soli 83MiB. Questa discrepanza è resa possibile dal fatto che il file system `ext4` può contenere [file sparsi](#).

Suggerimento

L'uso effettivo del disco dei [file sparsi](#) cresce insieme ai dati che in essi sono scritti.

Usando operazioni simili su device creati dal [device loop](#) o dal [device mapper](#), come in Sezione 9.7.3, si può partizionare tale immagine di disco "disk.img" usando `parted(8)` o `fdisk(8)` e si può creare in essa file system usando `mkfs.ext4(8)`, `mkswap(8)`, ecc.

9.7.6 Creare un file con un'immagine ISO9660

Si può creare un file immagine [ISO9660](#) "cd.iso" dell'albero di directory originale in "directory_sorgente" usando `genisoimage(1)` fornito da [cdrkit](#) nel modo seguente.

```
# genisoimage -r -J -T -V volume_id -o cd.iso source_directory
```

Analogamente, si può creare un file immagine ISO9660 avviabile, "cdboot.iso", da un albero di directory simile a quello del `debian-installer` in "directory_sorgente" nel modo seguente.

```
# 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
```

In questo esempio viene usato per l'avvio il [bootloader Isolinux](#) (vedere Sezione 3.1.2).

Si può calcolare il valore `md5sum` e creare l'immagine ISO9660 direttamente dal device CD-ROM nel modo seguente.

```
$ 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
```

**avvertimento**

Per ottenere un risultato corretto, si deve accuratamente evitare il bug di Linux riguardante il read ahead del file system ISO9660, come nell'esempio precedente.

9.7.7 Scrivere direttamente sul CD/DVD-R/RW

Suggerimento

Per `wodim(1)`, fornito da [cdrkit](#), un DVD è semplicemente un grande CD.

Si può trovare un device utilizzabile usando il comando seguente.

```
# wodim --devices
```

Poi si inserisce un CD-R vergine nell'unità CD e si scrive il file immagine ISO9660 "cd.iso" su questo device, ad esempio "/dev/hda", usando `wodim(1)` nel modo seguente.

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

Se viene usato un CD-RW invece di un CD-R, usare quest'altro comando.

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

Suggerimento

Se il sistema desktop usato monta automaticamente i CD, prima di usare `wodim(1)` smontarlo usando dalla console "sudo umount /dev/hda".

9.7.8 Montare un file con un'immagine ISO9660

Se "cd.iso" contiene un'immagine ISO9660, lo si può montare manualmente in "/cdrom" usando il comando seguente.

```
# mount -t iso9660 -o ro,loop cd.iso /cdrom
```

Suggerimento

I sistemi desktop moderni possono montare automaticamente i supporti removibili come i CD in formato ISO9660 (vedere Sezione [10.1.7](#)).

9.8 I dati binari

Questa sezione tratta della manipolazione diretta dei dati binari su supporti di archiviazione.

pacchetto	popcon	dimensione	descrizione
coreutils	V:901, I:999	17478	pacchetto base che contiene <code>od(1)</code> per fare il dump di file (HEX, ASCII, OCTAL, ...)
bsdmainutils	V:53, I:953	27	pacchetto di utilità che contiene <code>hd(1)</code> per fare il dump di file (HEX, ASCII, OCTAL, ...)
hexedit	V:1, I:12	73	visualizzatore ed editor binario (HEX, ASCII)
bless	V:0, I:4	1028	editor esadecimale completo (GNOME)
okteta	V:1, I:14	1504	editor esadecimale completo (KDE4)
ncurses-hexedit	V:0, I:2	132	visualizzatore ed editor binario (HEX, ASCII, EBCDIC)
beav	V:0, I:0	133	visualizzatore ed editor binario (HEX, ASCII, EBCDIC, OCTAL, ...)

Tabella 9.21: Elenco di pacchetti che visualizzano e modificano dati binari

9.8.1 Visualizzare e modificare dati binari

Il metodo di visualizzazione dei dati binari più basilare è l'uso del comando `"od -t x1"`.

Suggerimento

HEX è usato come acronimo per il formato [esadecimale](#) con [base 16](#). OCTAL è usato per il formato [ottale](#) con [base 8](#). ASCII è usato per [American Standard Code for Information Interchange](#), cioè la normale codifica per testi in inglese. EBCDIC è usato per [Extended Binary Coded Decimal Interchange Code](#) usato nei sistemi operativi [deimainframe IBM](#).

9.8.2 Manipolare file senza montare i dischi

Esistono strumenti per leggere e scrivere file senza montare i dischi.

pacchetto	popcon	dimensione	descrizione
mtools	V:9, I:82	399	utilità per file MSDOS senza montarli
hfsutils	V:0, I:6	184	utilità per file HFS e HFS+ senza montarli

Tabella 9.22: Elenco di pacchetti per manipolare file senza montare i dischi

9.8.3 Dati ridondanti

I sistemi software [RAID](#) offerti dal kernel Linux forniscono un livello di ridondanza dei dati nel file system a livello del kernel, allo scopo di ottenere una più alta affidabilità dell'archiviazione.

Esistono strumenti per aggiungere dati ridondanti a file a livello di programmi applicativi per ottenere anche in questo modo una più alta affidabilità dell'archiviazione.

pacchetto	popcon	dimensione	descrizione
par2	V:4, I:26	271	insiemi di volumi di archivi di parità, per controllare e riparare file
dvdaster	V:0, I:2	1741	protezione contro perdita di dati/graffi/invecchiamento di supporti CD/DVD
dvbackup	V:0, I:0	413	strumento di backup che usa camcorder MiniDV (fornisce <code>rsbep(1)</code>)
vdmfec	V:0, I:0	97	recupera blocchi perduti usando Forward Error Correction

Tabella 9.23: Elenco di strumenti per aggiungere dati ridondanti a file

9.8.4 recupero di file dati ed analisi forensi

Esistono strumenti per recuperare file dati e fare analisi forensi.

pacchetto	popcon	dimensione	descrizione
testdisk	V:3, I:37	1426	utilità per scansione delle partizione e ripristino di dischi
magicrescue	V:0, I:3	259	utilità per ripristinare file cercando byte magici
scalpel	V:0, I:4	87	strumento di escavazione di file frugale e ad alte prestazioni
myrescue	V:0, I:3	83	recupera dati da dischi fissi danneggiati
extundelete	V:0, I:11	147	utilità per decancellare file da file system ext3/4
ext4magic	V:0, I:5	233	utilità per decancellare file da file system ext3/4
ext3grep	V:0, I:3	293	strumento per aiutare a recuperare file cancellati da file system ext3
scrounge-ntfs	V:0, I:3	50	programma di recupero di dati da file system NTFS
gzrt	V:0, I:0	33	insieme di strumenti di recupero di gzip
sleuthkit	V:2, I:25	1511	strumenti per analisi forensi (Sleuthkit)
autopsy	V:0, I:1	1027	interfaccia grafica per SleuthKit
foremost	V:0, I:7	100	applicazione forense per il recupero dei dati
guymager	V:0, I:1	1035	strumento forense per immagini basato su Qt
dcfldd	V:0, I:5	106	versione migliorata di dd per analisi forensi e sicurezza

Tabella 9.24: Elenco di pacchetti per recupero di file dati ed analisi forensi.

Suggerimento

Si possono ripristinare i file sui file system ext2 usando i comandi `list_deletet_inodes` e `undel` di `debugfs(8)` nel pacchetto `e2fsprogs`.

9.8.5 Suddividere un file grande in file più piccoli

Quando dei dati sono troppo grandi affinché ne venga fatto il backup in un file singolo, si può fare il backup dei suoi contenuti dopo averlo suddiviso in pezzi di, ad esempio 2000MiB, che saranno successivamente riuniti a formare il file originale.

```
$ split -b 2000m large_file
$ cat x* >large_file
```



Attenzione

Assicurarsi di non avere altri file che iniziano con "x" per evitare conflitti nei nomi.

9.8.6 Pulire il contenuto di file

Per ripulire i contenuti di un file, come un file di registro, non usare `rm(1)` per cancellarlo e poi crearne un altro, perché nell'intervallo tra i due comandi potrebbe essere ancora possibile accedere al file. Quello che segue è il metodo sicuro per pulire il contenuto di un file.

```
$ :>file_to_be_cleared
```

9.8.7 File fittizi

I comandi seguenti creano file fittizi o vuoti.

```
$ 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
```

Dovrebbero essere ora presenti i seguenti file.

- "5kb.file" è costituito da 5KB di zero.
- "7mb.file" è costituito da 7MB di dati casuali.
- "zero.file" potrebbe essere un file di 0 byte. Se fosse stato preesistente, il suo orario `mtime` sarebbe stato aggiornato mentre sarebbero stati mantenuti i suoi contenuti e la sua grandezza.
- "semprezero.file" è sempre un file di 0 byte. Se fosse stato preesistente, il suo orario `mtime` sarebbe stato aggiornato e il suo contenuto azzerato.

9.8.8 Cancellare un intero disco fisso

Esistono diversi modi di cancellare completamente i dati da un intero device simile ad un disco fisso, ad esempio una chiavetta USB in `/dev/sda`.



Attenzione

Prima di eseguire i comandi indicati in seguito controllare la posizione della chiavetta USB con `mount(8)`. Il device a cui punta `/dev/sda` potrebbe essere il disco fisso SCSI o Serial-ATA che contiene l'intero sistema.

Cancellare tutti i contenuti del disco reimpostando tutti i dati a 0 con il comando seguente.

```
# dd if=/dev/zero of=/dev/sda
```

Cancellare tutto sovrascrivendo dati casuali con il comando seguente.

```
# dd if=/dev/urandom of=/dev/sda
```

Cancellare tutto sovrascrivendo dati casuali in modo molto efficiente con il comando seguente.

```
# shred -v -n 1 /dev/sda
```

Dato che `dd(1)` è disponibile dalla shell di molti CD Linux avviabili, come il CD dell'installatore Debian, si può cancellare completamente il sistema installato su un disco fisso, ad esempio `/dev/hda`, `/dev/sda`, ecc., eseguendo un comando di cancellazione da un supporto CD simile.

9.8.9 Cancellare area inutilizzate di un disco fisso

Area inutilizzate di un disco fisso (o di una chiavetta USB), ad esempio `/dev/sdb1`, potrebbero contenere ancora i dati cancellati stessi, dato che questi sono semplicemente scollegati dal file system. È possibile pulire queste aree sovrascrivendole.

```
# 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
```

**avvertimento**

Solitamente questo procedimento è sufficientemente buono per le chiavette USB. Ma non è perfetto. La maggior parte dei nomi di file cancellati e dei loro attributi potrebbe ancora essere nascosta e rimanere nel file system.

9.8.10 De-cancellare file cancellati ma ancora aperti

Anche se un file è stato cancellato per errore, fintanto che è usato da un'applicazione (in lettura o scrittura), è possibile recuperarlo.

Per esempio, provare a fare quanto segue.

```
$ 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
```

Quando si ha il pacchetto `lsOf` installato, eseguire in un altro terminale quanto segue.

```
$ 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 Cercare tutti i collegamenti fisici

I file con collegamenti fisici possono essere identificati usando `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
```

Entrambi `"pippo"` e `"pluto"` hanno il numero di collegamenti uguale a `"2"` (`>1`), il che mostra che hanno collegamenti fisici. Hanno il numero di `inode`, `"2738404"`, in comune. Ciò significa che sono lo stesso file con un collegamento fisico. Nell'eventualità che non si trovino tutti i file con collegamento fisico, li si possono cercare in base all'`inode`, ad esempio `"2738404"`, nel modo seguente.

```
# find /path/to/mount/point -xdev -inum 2738404
```

9.8.12 Consumo invisibile dello spazio su disco

Tutti i file cancellati ma aperti consumano spazio su disco, anche se non sono visibili con il normale comando `du(1)`. Possono essere elencati, insieme alla loro dimensione, nel modo seguente.

```
# lsof -s -X / |grep deleted
```

9.9 Suggerimenti per la cifratura dei dati

Avendo l'accesso fisico alla macchina, chiunque può facilmente ottenere privilegi di root e accedere a tutti i file nel PC (vedere Sezione 4.6.4). Ciò significa che il sistema delle password di login non può mettere al sicuro i dati privati e sensibili contro un possibile furto del PC. Per farlo deve essere usata una tecnologia di cifratura dei dati. Sebbene [GNU Privacy Guard](#) (vedere Sezione 10.3) possa cifrare file, richiede un po' di lavoro da parte dell'utente.

[Dm-crypt](#) facilita automatic data encryption via native Linux kernel modules with minimal user efforts using [device-mapper](#).

pacchetto	popcon	dimensione	descrizione
cryptsetup	V:41, I:77	407	utilità per device a blocchi cifrati (dm-crypt / LUKS)
cryptmount	V:4, I:5	228	utilità per device a blocchi cifrati (dm-crypt / LUKS) con particolare attenzione al montaggio/smontaggio da parte di utenti normali
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)

Tabella 9.25: Elenco di utilità per la cifratura dei dati



Attenzione

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

Nota

È possibile installare un intero file system Debian in un disco cifrato con l'[Installatore Debian](#) (lenny o successivo) usando [dm-crypt/LUKS](#) e [initramfs](#).

Suggerimento

Per lo strumento di cifratura in spazio utente [GNU Privacy Guard](#) vedere Sezione 10.3.

9.9.1 Cifratura di dischi removibili con dm-crypt/LUKS

Si può cifrare il contenuto di dispositivi di memorizzazione di massa removibili, ad esempio una chiavetta USB in `"/dev/sdx"`, usando [dm-crypt/LUKS](#). Formattarla semplicemente nel modo seguente.

```
# 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 Sezione 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 Cifratura della partizione di swap con dm-crypt

In questo esempio si suppone che il file `"/etc/fstab"` originale contenga quanto segue.

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

Si può abilitare la cifratura della partizione di swap usando `dm-crypt` nel modo seguente.

```
# 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 Montare dischi removibili con dm-crypt/LUKS

Una partizione di disco cifrata creata con `dm-crypt/LUKS` su `"/dev/sdc5"` può essere montata in `"/mnt"` nel modo seguente:

```
$ 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 Il kernel

Debian distribuisce, per le architetture supportate, kernel Linux modulari contenuti in pacchetti.

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

9.10.1 Parametri del kernel

Molte caratteristiche di Linux possono essere configurate tramite parametri del kernel, nei modi seguenti.

- Parametri del kernel inizializzati dal bootloader (vedere Sezione 3.1.2)
- Se accessibili attraverso sysfs (vedere Sezione 1.2.12), parametri del kernel modificati da `sysctl(8)` durante l'esecuzione del sistema.
- Parametri di moduli impostati dagli argomenti di `modprobe(8)` al momento dell'attivazione di un modulo (vedere Sezione 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 Header del kernel

La maggior parte dei **normali programmi** non ha bisogno degli header del kernel per essere compilata, anzi di fatto può corrompersi se si usano direttamente gli header. Questi programmi dovrebbero essere compilati nel sistema Debian usando gli header in `"/usr/include/linux"` e `"/usr/include/asm"` forniti dal pacchetto `libc6-dev` (creato dal pacchetto sorgente `glibc`).

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 Compilare il kernel ed i moduli relativi

Debian ha un proprio metodo di compilazione del kernel e dei moduli relativi.

pacchetto	popcon	dimensione	descrizione
build-essential	I:500	20	pacchetti essenziali per compilare pacchetti Debian: <code>make</code> , <code>gcc</code> , ...
bzip2	V:147, I:973	122	utilità di compressione e decompressione per file <code>bz2</code>
libncurses5-dev	I:113	6	librerie di sviluppo e documentazione per <code>ncurses</code>
git	V:312, I:494	36515	<code>git</code> : sistema distribuito di controllo delle revisioni usato dal kernel Linux
fakeroot	V:32, I:518	228	fornisce un ambiente <code>fakeroot</code> per compilare pacchetti da utente non root
initramfs-tools	V:442, I:989	112	strumento per compilare un <code>initramfs</code> (specifico di Debian)
dkms	V:99, I:210	295	DKMS (Dynamic Kernel Module Support, supporto dinamico per i moduli del kernel) (generico)
module-assistant	V:1, I:30	391	helper tool to make module package (Debian specific)
devscripts	V:8, I:54	2654	script di aiuto per i manutentori di pacchetti Debian (specifico di Debian)

Tabella 9.26: Elenco di pacchetti chiave da installare per la ricompilazione del kernel in un sistema Debian

Se si usa `initrd` nello Sezione 3.1.2, ci si assicuri di leggere le informazioni relative in `initramfs-tools(8)`, `update-initramfs(8)` e `initramfs.conf(5)`.

**avvertimento**

Quando si compilano i sorgenti del kernel Linux, non mettere collegamenti simbolici alle directory nell'albero dei sorgenti (ad esempio, `"/usr/src/linux*"`) in `"/usr/include/linux"` e `"/usr/include/asm"`. (Alcuni documenti ormai datati suggeriscono di farlo.)

Nota

Quando si compila il kernel Linux più recente nel sistema Debian stable, potrebbe essere necessario l'uso delle versioni backport degli strumenti più recenti da 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.

Il [DKMS \(Dynamic Kernel Module Support, supporto dinamico per i moduli del kernel\)](#) è una nuova infrastruttura indipendente dalla distribuzione progettata per permettere l'aggiornamento di singoli moduli del kernel senza cambiare tutto il kernel. È utilizzata per il mantenimento dei moduli esterni all'albero dei sorgenti. Rende anche molto facile la ricompilazione dei moduli quando si aggiornano i kernel.

9.10.4 Compilare i sorgenti del kernel: il metodo raccomandato dal Team del Kernel di Debian

Per compilare pacchetti binari del kernel personalizzati a partire dai sorgenti originali del kernel, si deve usare il target `"deb-pkg"` fornito.

```
$ 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
```

Suggerimento

Il pacchetto `linux-source-versione` fornisce i sorgenti del kernel Linux con le patch Debian come `"/usr/src/linux-versione.tar.bz2"`.

Per compilare pacchetti binari specifici a partire dal pacchetto Debian dei sorgenti del kernel, si devono usare i target `"binary-arch_ari"` in `"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
```

Ulteriori informazioni:

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

9.10.5 Driver per 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.

- driver [GPU](#)
 - driver GPU Intel (`main`)
 - driver GPU AMD/ATI (`main`)
 - driver GPU NVIDIA (`main` per il driver [nouveau](#) e `non-free` per i driver solo binari supportati dal produttore).
- driver [Softmodem](#)
 - pacchetti `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`

Notare che i pacchetti `non-free` e `contrib` non fanno parte del sistema Debian. La configurazione per abilitare e disabilitare le aree `non-free` e `contrib` è descritta in Sezione [2.1.4](#). Si dovrebbe essere consapevoli degli aspetti negativi legati all'uso dei pacchetti `non-free` e `contrib` come descritto in Sezione [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 virtualizzato

L'uso di sistemi virtualizzati permette di eseguire più istanze di un sistema simultaneamente su un singolo hardware.

Suggerimento

Vedere <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 Sezione [4.7.4](#) and it is the backend technology of Sezione [7.6](#).

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

pacchetto	popcon	dimensione	descrizione
schroot	V:7, I:9	2708	strumento specializzato per eseguire pacchetti Debian binari in chroot
sbuid	V:1, I:4	271	strumento per compilare pacchetti Debian binari da sorgenti Debian
debootstrap	V:5, I:62	298	avviare un sistema Debian base (scritto in sh)
cdebootstrap	V:0, I:2	116	avviare un sistema Debian (scritto in C)
virt-manager	V:10, I:43	2298	Virtual Machine Manager : applicazione desktop per gestire macchine virtuali
libvirt-clients	V:46, I:64	1130	programmi per la libreria libvirt
games-emulator	I:0	26	games-emulator : Debian's emulators for games
bochs	V:0, I:1	7194	Bochs : emulatore PC IA-32
qemu	I:31	96	QEMU : veloce emulatore generico di processore
qemu-system	I:21	97	QEMU : binari per emulazione completa del sistema
qemu-user	V:0, I:11	89683	QEMU : binari per emulazione in spazio utente
qemu-utils	V:11, I:108	6077	QEMU : utilità
qemu-kvm	V:8, I:55	107	KVM : virtualizzazione completa su hardware x86 con virtualizzazione assistita da hardware
virtualbox	V:11, I:14	107018	VirtualBox : soluzione per virtualizzazione i dx86 su i386 e amd64
xen-tools	V:0, I:4	727	strumenti per gestire server virtuali XEN Debian
wine	V:16, I:76	191	Wine : implementazione della API Windows (suite standard)
dosbox	V:2, I:18	2702	DOSBox : emulatore x86 con grafica Tandy/Herc/CGA/EGA/VGA/SVGA, suono e DOS
dosemu	V:0, I:2	4891	DOSEMU : emulatore DOS di Linux
lxc	V:11, I:14	18971	strumenti in spazio utente per contenitori 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

Tabella 9.27: Elenco di strumenti di virtualizzazione

Vedere l'articolo di Wikipedia [Comparison of platform virtual machines](#) per una comparazione dettagliata di diverse soluzioni per la virtualizzazione di piattaforme.

9.11.2 Fasi del processo di virtualizzazione

Nota

I kernel Debian predefiniti hanno il supporto per [KVM](#) a partire da Lenny.

Il tipico processo di [virtualizzazione](#) comporta diverse fasi.

- Creare un file system vuoto (un albero di file o un'immagine di disco).
 - L'albero di file può essere creato con `mkdir -p /percorso/di/chroot`.
 - Il file immagine raw del disco può essere creato con `dd(1)` (vedere Sezione [9.7.1](#) e Sezione [9.7.5](#)).
 - Si può usare `qemu-img(1)` per creare e convertire file immagine di dischi supportati da [QEMU](#).
 - I formati di file raw e [VMDK](#) possono essere usati come formati comuni tra i vari strumenti di virtualizzazione.
- Montare l'immagine del disco con `mount(8)` nel file system (opzionale).
 - Per il file immagine raw del disco, montarlo come [device loop](#) o [device mapper](#) (vedere Sezione [9.7.3](#)).
 - Per le immagini disco supportate da [QEMU](#), montarle come [device a blocchi di rete](#) (vedere Sezione [9.11.3](#)).
- Popolare il file system obiettivo con i dati di sistema necessari.
 - L'uso di programmi come `debootstrap` e `cdebootstrap` aiuta questo processo (vedere Sezione [9.11.4](#)).
 - Usare gli installatori di sistemi operativi nell'emulazione di sistemi completi.
- Eseguire un programma in un ambiente virtualizzato.
 - [chroot](#) fornisce un ambiente di virtualizzazione base sufficiente a compilare programmi, eseguire applicazioni in console ed eseguire demoni al suo interno.
 - [QEMU](#) fornisce emulazione di CPU inter-piattaforma.
 - [QEMU](#) con [KVM](#) fornisce una completa emulazione di sistema con [virtualizzazione assistita da hardware](#).
 - [VirtualBox](#) fornisce una completa emulazione del sistema in i386 e amd64 con o senza [virtualizzazione assistita da hardware](#).

9.11.3 Montare il file immagine di disco virtuale

Per i file immagine raw di disco, vedere Sezione [9.7](#).

Per altri file immagine di dischi virtuali, si può usare `qemu-nbd(8)` per esportarli usando il protocollo per [device a blocchi di rete](#) e montarli usando il modulo `nbd` del kernel.

`qemu-nbd(8)` supporta i formati di disco supportati da [QEMU](#); [QEMU](#) supporta i seguenti formati di dischi: `raw`, `qcow2`, `qcow`, `vmdk`, `vdi`, `bochs`, `cow` (copy-on-write di user-mode Linux), `parallels`, `dmg`, `cloop`, `vpc`, `vfat` (VFAT virtuale) e `host_device`.

I [device a blocchi di rete](#) possono supportare partizioni nello stesso modo dei [device loop](#) (vedere Sezione [9.7.3](#)). Si può montare la prima partizione di `"disk.img"` nel modo seguente.

```
# modprobe nbd max_part=16
# qemu-nbd -v -c /dev/nbd0 disk.img
...
# mkdir /mnt/part1
# mount /dev/nbd0p1 /mnt/part1
```

Suggerimento

È possibile esportare solamente la prima partizione di `"disk.img"` usando l'opzione `"-P 1"` per `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.

**Attenzione**

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.

Suggerimento

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 Sistemi desktop multipli

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.

L'immagine disco virtuale `"virtdisk.qcow2"` contenente un sistema Debian per [QEMU](#) si può creare con i [piccoli CD dell'installatore Debian](#) nel modo seguente.

```
$ 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
...
```

Vedere altri suggerimenti su [Debian Wiki: QEMU](#).

[VirtualBox](#) è fornito con strumenti con interfaccia utente grafica [Qt](#) piuttosto intuitivi. I suoi strumenti grafici e a riga di comando sono spiegati nel [Manuale utente di VirtualBox](#) e [Manuale utente di VirtualBox \(PDF\)](#).

Suggerimento

Eseguire altre distribuzioni GNU/Linux come [Ubuntu](#) e [Fedora](#) in una [virtualizzazione](#) è un ottimo metodo per imparare trucchetti di configurazione. Anche altri sistemi operativi proprietari possono essere eseguiti tranquillamente in queste [virtualizzazioni](#) GNU/Linux.

Capitolo 10

Gestione dei dati

In questo capitolo sono descritti strumenti e trucchi per gestire dati binari e di testo in un sistema Debian.

10.1 Condividere, copiare ed archiviare

**avvertimento**

L'accesso in scrittura non coordinato a device a cui si sta attivamente accedendo e a file da parte di processi diversi deve essere evitato per prevenire le [race condition](#). Per evitare ciò devono essere usati i meccanismi di [lock dei file](#) utilizzando `flock(1)`.

La sicurezza dei dati e la loro condivisione controllata hanno diversi aspetti.

- La creazione di archivi di dati
- L'accesso ad archivi remoti
- La duplicazione
- Il tenere traccia della cronologia delle modifiche
- La facilitazione della condivisione dei dati
- Il prevenire l'accesso non autorizzato ai file
- La rilevazione di modifiche non autorizzate ai file

Queste azioni possono essere realizzate usando una combinazione di strumenti.

- Strumenti di archiviazione e compressione
 - Strumenti di copia e sincronizzazione
 - file system di rete
 - Supporti di archiviazione removibili
 - Secure Shell
 - Il sistema di autenticazione
 - Strumenti per sistemi di controllo delle versioni
 - Strumenti crittografici per hash e cifratura
-

10.1.1 Strumenti di archiviazione e compressione

Ecco una tabella riassuntiva degli strumenti di archiviazione e compressione disponibili per il sistema Debian.

**avvertimento**

Non impostare la variabile "\$TAPE" a meno che non si sappia esattamente cosa aspettarsi. Cambia il comportamento di `tar(1)`.

- Gli archivi `tar(1)` compressi con `gzip` usano l'estensione di file ".tgz" o ".tar.gz".
- Gli archivi `tar(1)` compressi con `xz` usano l'estensione di file ".txz" o ".tar.xz".
- La popolarità dei metodi di compressione negli strumenti FOSS come `tar(1)` è cambiata nel tempo nel modo seguente `gzip` → `bzip2` → `xz`
- `cp(1)`, `scp(1)` e `tar(1)` possono avere alcune limitazioni per file speciali. `cpio(1)` è più versatile.
- `cpio(1)` è progettato per essere usato con `find(1)` ed altri comandi adatti per creare script di backup, dato che la porzione di selezione dei file dello script può essere testata in modo autonomo.
- La struttura interna dei file di dati di LibreOffice è quella dei file «.jar» che può essere aperta anche da `unzip`.
- Lo strumento multiplatforma di fatto usato per gli archivi è `zip`. Usarlo come «`zip -rX`» per ottenere la massima compatibilità. Usare anche l'opzione «-S» se è importante la dimensione massima dei file.

10.1.2 Strumenti di copia e sincronizzazione

Ecco una tabella riassuntiva dei semplici strumenti di copia e backup disponibili in un sistema Debian.

La copia dei file con `rsync(8)` offre un insieme di funzionalità più ricco di altri strumenti.

- l'algoritmo delta-transfer, che invia solamente le differenze tra il file sorgente ed il file esistente nella destinazione
- algoritmo veloce di verifica (predefinito) che cerca i file la cui dimensione o il cui orario di ultima modifica sono cambiati
- opzioni "- -exclude" e "- -exclude-from" simili a quelle di `tar(1)`
- sintassi con "una barra / alla fine della directory sorgente" che evita di dover creare un livello aggiuntivo di directory nella destinazione

Suggerimento

Gli strumenti di controllo delle versioni VCS (Version control system) in Tabella 10.14 possono essere usati come strumenti di copia e sincronizzazione multidirezionali.

10.1.3 Esempi di invocazione per archivi

Ecco diversi modi di archiviare ed estrarre archivi con l'intero contenuto della directory "./sorgente", usando diversi strumenti.

GNU `tar(1)`:

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

In alternativa usare i comandi seguenti.

pacchetto	popcon	dimensione	estensione	comando	commento
tar	V:914, I:999	3152	.tar	tar(1)	strumento di archiviazione standard (standard de facto)
cpio	V:489, I:998	1144	.cpio	cpio(1)	strumento di archiviazione Unix in stile System V, da usare con find(1)
binutils	V:164, I:673	97	.ar	ar(1)	strumento di archiviazione per la creazione di librerie statiche
fastjar	V:2, I:25	183	.jar	fastjar(1)	strumento di archiviazione per Java (simile a zip)
pax	V:12, I:24	170	.pax	pax(1)	nuovo strumento POSIX di archiviazione, compromesso tra tar e cpio
gzip	V:891, I:999	242	.gz	gzip(1) , zcat(1) , ...	utilità GNU di compressione LZ77 (standard de facto)
bzip2	V:147, I:973	122	.bz2	bzip2(1) , bzcat(1) , ...	utilità per compressione con ordinamento dei blocchi Burrows-Wheeler con maggiore rapporto di compressione di gzip(1) (più lenta di gzip con sintassi simile)
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) , ...	utilità di compressione XZ con maggiore rapporto di compressione di bzip2(1) (più lenta di gzip , ma più veloce di bzip2 ; sostituto dell'utilità di compressione 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)	strumento di archiviazione file 7-Zip con alto rapporto di compressione (compressione LZMA)
p7zip-full	V:102, I:469	4664	.7z	7z(1) , 7za(1)	strumento di archiviazione file 7-Zip con alto rapporto di compressione (compressione LZMA e altre)
lzop	V:10, I:85	164	.lzo	lzop(1)	utilità di compressione LZO con velocità di compressione e decompressione più alta di quella di gzip(1) (più basso rapporto di compressione di gzip con sintassi simile)
zip	V:49, I:427	623	.zip	zip(1)	InfoZIP : strumento di archiviazione e compressione per DOS
unzip	V:132, I:792	385	.zip	unzip(1)	InfoZIP : strumento di estrazione di archivi e decompressione per DOS

Tabella 10.1: Elenco di strumenti di archiviazione e compressione

pacchetto	popcon	dimensione	strumento	funzione
coreutils	V:901, I:999	17478	GNU cp	copia file e directory localmente ("a" per modalità ricorsiva)
openssh-client	V:809, I:997	4298	scp	copia file e directory da remoto (client, "r" per modalità ricorsiva)
openssh-server	V:701, I:844	1567	sshd	copia file e directory da remoto (server remoto)
rsync	V:311, I:564	677		sincronizzazione e backup in remoto unidirezionale
unison	V:3, I:16	14		sincronizzazione e backup in remoto bidirezionale

Tabella 10.2: Elenco di strumenti di copia e sincronizzazione

```
$ 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
```

10.1.4 Esempi di invocazione per la copia

Ecco diversi modi di copiare l'intero contenuto della directory `"/sorgente"`, usando diversi strumenti.

- Copia locale: directory `"/sorgente"` → directory `"/dest"`
- Copia remota: directory `"/sorgente"` sull'host locale → directory `"/dest"` sull'host `"utente@host.dom"`

rsync(8):

```
# cd ./source; rsync -aHAXSv . /dest
# cd ./source; rsync -aHAXSv . user@host.dom:/dest
```

In alternativa si può usare la sintassi con `"una barra / alla fine della directory sorgente"`.

```
# rsync -aHAXSv ./source/ /dest
# rsync -aHAXSv ./source/ user@host.dom:/dest
```

In alternativa usare i comandi seguenti.

```
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . /dest
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . user@host.dom:/dest
```

GNU cp(1) e openSSH scp(1):

```
# 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
```

In tutti gli esempi contenenti ".", si può sostituire "." con "pippo" per copiare i file dalla directory "./sorgente/pippo" alla directory "/dest/pippo".

In tutti gli esempi contenenti ".", si può sostituire "." con il percorso assoluto "/percorso/di/sorgente/pippo" per evitare di fare "cd ./sorgente;". I file verranno copiati in posizioni diverse a seconda dello strumento utilizzato, come descritto qui di seguito.

- in "/dest/pippo": `rsync(8)`, `GNU cp(1)` e `scp(1)`
- "/dest/percorso/di/sorgente/pippo": `GNU tar(1)` e `cpio(1)`

Suggerimento

`rsync(8)` e `GNU cp(1)` hanno l'opzione "-u" per saltare i file che sono più recenti nella destinazione.

10.1.5 Esempi di invocazione per la selezione di file

`find(1)` viene usato per selezionare i file per i comandi di archiviazione e copia (vedere Sezione 10.1.3 e Sezione 10.1.4) o per `xargs(1)` (vedere Sezione 9.4.9). Questo funzionamento può essere migliorato usando le sue opzioni di comando.

La sintassi base di `find(1)` può essere riassunta nel modo seguente.

- Gli argomenti condizionali sono valutati da sinistra a destra.
- Questa valutazione si ferma una volta che il risultato è determinato.
- L'operatore "OR logico" (specificato con "-o" tra condizioni) ha una precedenza più bassa dell'operatore "AND logico" (specificato da "-a" o dall'assenza di un operatore tra condizioni).
- L'operatore "NOT logico" (specificato da "!" prima di una condizione) ha una precedenza più alta di un operatore "AND logico".
- L'opzione "-prune" restituisce sempre una condizione di **VERO** logico e, se si tratta di una directory, la ricerca si ferma a questo punto.
- L'opzione "-name" trova corrispondenze con il nome base del file tramite espressioni glob di shell (vedere Sezione 1.5.6), ma fa corrispondenza anche con il carattere iniziale "." con l'uso di metacaratteri come "*" e "?". (Nuova funzionalità [POSIX](#).)
- L'opzione "-regex" trova corrispondenze con il percorso completo usando, in modo predefinito, **BRE** in stile emacs (vedere Sezione 1.6.2).
- L'opzione "-size" trova corrispondenze con file in base alla loro dimensione (valori preceduti da "+" o "-" per cercare dimensioni, rispettivamente, più grandi o piccole del valore).
- L'opzione "-newer" trova corrispondenze con file più recenti di quello specificato come argomento dell'opzione.
- L'opzione "-print0" restituisce sempre il valore logico **VERO** e stampa il nome file completo ([terminato dal carattere null](#)) sullo standard output.

`find(1)` è spesso usato con uno stile di invocazione come il seguente.

```
# 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
```

Il comando precedente si traduce nelle azioni seguenti.

1. Cercare tutti i file a partire da `"/percorso/di"`
2. Limitare globalmente la ricerca al file system da cui è richiamato e usare espressioni regolari **ERE** (vedere Sezione 1.6.2)
3. Escludere i file che corrispondono all'espressione regolare `".*\ .cpio" o ".*~"` dalla ricerca fermando la loro elaborazione
4. Escludere le directory che corrispondono all'espressione regolare `".*/\ .git"` dalla ricerca fermando la loro elaborazione
5. Escludere i file più grandi di 99 Megabyte (unità di 1048576 byte) dalla ricerca fermando la loro elaborazione
6. Stampare i nomi di file che soddisfano le condizioni di ricerca precedenti e che siano più recenti di `"/percorso/di/marcatura"`

Notare nell'esempio precedente l'uso della parte di comando `"-prune -o` per escludere file.

Nota

Alcune opzioni per `find(1)` potrebbero non essere supportate per i sistemi **nix* non Debian. In questi casi, cambiare le invocazioni in quelle adatte corrispondenti e sostituire `"-print0"` con `"-print"`. Potrebbe essere necessario modificare anche comandi correlati.

10.1.6 Supporti di archiviazione

Quando si deve scegliere il [supporto di archiviazione di dati informatici](#) per un importante archivio di dati, si dovrebbe porre attenzione alle limitazioni dei supporti. Per piccoli backup di dati personali, io uso CD-R e DVD-R scegliendoli in base alla marca del produttore e conservandoli in un ambiente fresco, all'ombra, asciutto e pulito. (I supporti di archiviazione a nastro sembrano molto popolari per gli usi professionali.)

Nota

Le [casseforti a prova di fuoco](#) sono pensate per i documenti cartacei. La maggior parte dei supporti di archiviazione di dati informatici ha una tolleranza più bassa alle alte temperature rispetto alla carta. Di solito mi affido a copie multiple cifrate sicure conservate in diverse posizioni sicure.

Durata di vita ottimistica di diversi supporti di archiviazione da dati raccolti in rete (per lo più dalle informazioni dei produttori).

- 100+ anni: carta senza acidi con inchiostro
- 100 anni: supporti ottici (CD/DVD, CD/DVD-R)
- 30 anni: supporti magnetici (nastri, dischetti floppy)
- 20 anni: supporti ottici a cambio di fase (CD-RW)

Questi tempi non tengono conto dei danni meccanici causati dal maneggiamento, ecc.

Cicli di scrittura ottimistici di diversi supporti di archiviazione da dati raccolti in rete (per lo più dalle informazioni dei produttori).

- 250,000+ cicli: unità a disco fisso
 - 10,000+ cicli: memoria flash
 - 1,000 cicli: CD/DVD-RW
 - 1 ciclo: CD/DVD-R, carta
-

**Attenzione**

I valori di durata di vita e dei cicli di scrittura riportati non dovrebbero essere usati per prendere decisioni riguardo all'archiviazione di dati critici. Consultare le informazioni specifiche per ciascun prodotto forniti dal produttore.

Suggerimento

Dato che i CD/DVD-R e la carta hanno un solo ciclo di scrittura, prevengono per loro stessa natura le perdite accidentali di dati per sovrascrittura. Questo è un vantaggio!

Suggerimento

Se è necessario fare backup frequenti e veloci di una grande quantità di dati, un disco fisso su un host remoto connesso con una connessione veloce, potrebbe essere l'unica soluzione realistica.

10.1.7 Supporti di archiviazione removibili

Un support di archiviazione removibile può essere uno dei seguenti.

- [Unità flash USB](#)
- [Unità a disco fisso](#)
- [Unità a dischi ottici](#)
- Fotocamera digitale
- Lettore audio digitale

Possono essere connessi in uno dei modi seguenti.

- [USB](#)
- [IEEE 1394 / FireWire](#)
- [Scheda PC](#)

Gli ambienti desktop moderni, come GNOME e KDE, possono montare questi dispositivi removibili automaticamente senza una voce corrispondente in `"/etc/fstab"`.

- [udisks2](#) package provides a daemon and associated utilities to mount and unmount these devices.
- [D-bus](#) crea eventi per dare inizio ai processi automatici.
- [PolicyKit](#) fornisce i privilegi necessari.

Suggerimento

I dispositivi montati automaticamente hanno l'opzione di mount `"uhide lper="` che viene usata da `umount(8)`.

Suggerimento

Nei moderni ambienti desktop il montaggio automatico avviene solo quando i device dei supporti removibili non sono elencati in `"/etc/fstab"`.

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

- `mlabel(1)` per file system FAT
- `genisoimage(1)` con l'opzione `"-V"` per file system ISO9660
- `tune2fs(1)` con l'opzione `"-L"` per file system ext2/ext3/ext4

Suggerimento

Può essere necessario fornire come opzione di montaggio la scelta della codifica (vedere Sezione [8.1.3](#)).

Suggerimento

L'uso del menu GUI per smontare un file system può rimuovere il suo nodo di device generato dinamicamente come `"/dev/sdc"`. Se si desidera mantenere il suo nodo di device, smontarlo con il comando `umount(8)` dal prompt di shell.

10.1.8 Scelta del file system per la condivisione di dati

Quando si condividono dati con un altro sistema attraverso dispositivi di archiviazione removibili, quest'ultimi andrebbero formattati con un [filesystem](#) comune supportato da entrambi i sistemi. Quello che segue è un elenco delle scelte possibili per il file system.

filesystem name	typical usage scenario
FAT12	condivisione interpiattaforma di dati su dischetti floppy (<32MiB)
FAT16	condivisione interpiattaforma di dati su dispositivi come piccoli dischi fissi (<2GiB)
FAT32	condivisione interpiattaforma di dati su dispositivi come grandi dischi fissi (<8TiB, supportato da sistemi più recenti di 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	condivisione interpiattaforma di dati su dispositivi come grandi dischi fissi (supportato nativamente su MS Windows NT e versioni successive e supportato da NTFS-3G attraverso FUSE in Linux)
ISO9660	condivisione interpiattaforma di dati statici su CD-R e DVD+/-R
UDF	scrittura incrementale di dati su CD-R e DVD+/-R (nuovo)
MINIX	archiviazione, efficiente in termini di spazio, di file dati unix su dischetti floppy
ext2	condivisione di dati su dispositivi come dischi fissi con sistemi Linux più vecchi
ext3	condivisione di dati su dispositivi come dischi fissi con sistemi Linux più vecchi
ext4	condivisione di dati su dispositivi come dischi fissi con sistemi Linux recenti
btrfs	sharing of data on the hard disk like device with current Linux systems with read-only snapshots

Tabella 10.3: Elenco di possibili scelte per il file system di dispositivi di archiviazione removibili con scenari di uso tipici

Suggerimento

Vedere Sezione [9.9.1](#) per la condivisione interpiattaforma di dati usando cifratura a livello di dispositivo.

Il file system FAT è supportato da quasi tutti i sistemi operativi moderni ed è piuttosto utile per scopi di scambio di dati attraverso supporti come dischi fissi removibili.

Quando si formatta un dispositivo come un disco fisso removibile con il file system FAT per la condivisione interpiattaforma di dati, le scelte seguenti dovrebbero essere quelle più sicure.

- Partizionare con `fdisk(8)`, `cdisk(8)` o `parted(8)` (vedere Sezione 9.6.2) creando un'unica partizione primaria e marcarla nel modo seguente.
 - Tipo "6", cioè FAT16, per supporti più piccoli di 2GB.
 - Tipo "c", cioè FAT32 (LBA), per supporti più grandi.
- Formattare la partizione primaria con `mkfs.vfat(8)` nel modo seguente.
 - Per FAT16, semplicemente con il suo nome di device, ad esempio `"/dev/sda1"`.
 - Per FAT32, con il suo nome di device e l'opzione esplicita, ad esempio `"-F 32 /dev/sda1"`

Quando si usano i file system FAT o ISO9660 per la condivisione dei dati, per essere sicuri dei risultati, si dovrebbero considerare i seguenti aspetti.

- Archiviare prima i file in un file di archivio usando `tar(1)` o `cpio(1)` per mantenere i nomi di file lunghi, i collegamenti simbolici, i permessi Unix sui file originali e le informazioni sui proprietari.
- Suddividere il file di archivio in pezzi più piccoli di 2 GiB con il comando `split(1)` per proteggerli da limitazioni sulla dimensione dei file.
- Cifrare il file archivio per proteggere i suoi contenuti da accesso non autorizzato.

Nota

Il file system FAT, per sua stessa natura, permette una dimensione massima per i file di $(2^{32} - 1)$ byte = (4GiB - 1 byte). Per alcune applicazioni su sistemi operativi a 32 bit più vecchi, la dimensione massima per i file era ancora più piccola: $(2^{31} - 1)$ byte = (2GiB - 1 byte). Debian non soffre di quest'ultima limitazione.

Nota

La stessa Microsoft non raccomanda l'uso di FAT per le unità o le partizioni più grandi di 200 MB. Microsoft evidenzia le sue limitazioni, quali un uso inefficiente dello spazio su disco, nel documento "[Overview of FAT, HPFS, and NTFS File Systems](#)". Naturalmente per Linux si dovrebbe normalmente usare il file system ext4.

Suggerimento

Per maggiori informazioni sui file system e sull'accesso ad essi, leggere il "[Filesystems HOWTO](#)".

10.1.9 Condividere dati attraverso una rete

Quando si condividono dati con un altro sistema attraverso una rete, si dovrebbero tenere a mente i servizi comuni. Ecco alcuni suggerimenti.

Sebbene questi file system montati in rete e metodi di trasferimento di file attraverso la rete siano piuttosto comodi per la condivisione dei dati, possono essere non sicuri. La loro connessione di rete deve essere resa sicura nel modo seguente.

- Cifrarli con [SSL/TLS](#)
- Usarli in tunnel [SSH](#)
- Usarli in tunnel [VPN](#)
- Limitarli dietro ad un firewall sicuro

Vedere anche Sezione 6.5 e Sezione 6.6.

servizio di rete	descrizione dello scenario di uso tipico
file system montato di rete SMB/CIFS con Samba	condivisione di file attraverso "rete Microsoft Windows", vedere <code>smb.conf(5)</code> e The Official Samba 3.x.x HOWTO and Reference Guide o il pacchetto <code>samba-doc</code>
file system montato di rete NFS con il kernel Linux	condivisione di file attraverso "rete Unix/Linux", vedere <code>exports(5)</code> e Linux NFS-HOWTO
servizio HTTP	condivisione di file tra client/server web
servizio HTTPS	condivisione di file tra client/server web con SSL (Secure Sockets Layer) cifrato o TLS (Transport Layer Security)
servizio FTP	condivisione di file tra client/server FTP

Tabella 10.4: Elenco dei servizi di rete da scegliere in base allo scenario di uso tipico

10.2 Backup e ripristino

Tutti sanno che i computer a volte si danneggiano oppure errori umani causano danni al sistema e ai dati. Le operazioni di backup e ripristino sono una parte essenziale di un'amministrazione di sistema di successo. Tutte i possibili modi in cui si possono creare danni si verificano prima o poi.

Suggerimento

Mantenere il proprio sistema di backup semplice e fare il backup di sistema spesso. Avere dati di backup è più importante della qualità tecnica del metodo di backup.

Ci sono 3 fattori chiave che determinano la reale politica di backup e ripristino.

1. Sapere di cosa fare il backup ed il ripristino

- I file dati direttamente creati dall'utente: in `~/`
- I file dati creati da applicazioni usate dall'utente: dati in `/var/` (tranne `/var/cache/`, `/var/run/` e `/var/tmp/`)
- File di configurazione del sistema: dati in `/etc/`
- Software locali: dati in `/usr/local/` o `/opt/`
- Informazioni di installazione del sistema: un memorandum in puro testo sui passi chiave (partizioni, ...)
- Insiemi di dati comprovati: confermati da operazioni preventive sperimentali di ripristino

2. Sapere come fare il backup ed il ripristino

- Rendere sicura l'archiviazione dei dati: protezione da sovrascritture e fallimenti del sistema
- Backup frequenti: backup pianificati
- Backup ridondanti: mirror di dati
- Procedura a prova di idioti: singolo facile comando di backup

3. Valutazione dei rischi e dei costi

- Valore dei dati se persi
- Risorse necessarie per il backup: umane, hardware, software, ...
- Modi in cui le cose possono andare storte e loro probabilità

Nota

Non fare il backup dei contenuti dei pseudo file system che si trovano in `/proc`, `/sys`, `/tmp` e `/run` (vedere Sezione [1.2.12](#) e Sezione [1.2.13](#)). A meno di non sapere esattamente ciò che si sta facendo, sono un'enorme mole di dati senza utilità.

Per quanto riguarda il rendere sicura l'archiviazione dei dati, questi dovrebbero essere come minimo in una partizione diversa del disco e preferibilmente su dischi e macchine diversi per sopravvivere alla corruzione del file system. È preferibile archiviare i dati importanti su supporti scrivibili una sola volta, come CD/DVD-R per prevenire incidenti di sovrascrittura. (Vedere Sezione 9.8 per come scrivere sul supporto di archiviazione dalla riga di comando. L'ambiente desktop GNOME con interfaccia grafica fornisce un facile accesso tramite menu: "Risorse → Creazione CD/DVD".)

Nota

Durante il backup dei dati può essere preferibile fermare alcuni demoni applicativi come l'MTA (vedere Sezione 6.2.4).

Nota

Si dovrebbe dare un'attenzione particolare al backup ed al ripristino di file con dati relativi all'identità dell'utente, come `/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`. Alcuni di questi dati non possono essere ricreati inserendo la stessa stringa di input nel sistema.

Nota

Se si esegue un compito di cron come processo di un utente, si deve ripristinare i file nella directory `/var/spool/cron/crontabs` e riavviare `cron(8)`. Vedere Sezione 9.4.14 per informazioni su `cron(8)` e `crontab(1)`.

10.2.1 Suite con utilità di backup

Quello che segue è un elenco di importanti suite di utilità di backup disponibili in un sistema Debian

Gli strumenti di backup hanno una propria specializzazione.

- [Mondo Rescue](#) è un sistema di backup per facilitare il ripristino veloce di un sistema completo a partire da backup su CD/DVD ecc., senza dover affrontare il normale processo di installazione del sistema.
- [Bacula](#), [Amanda](#) e [BackupPC](#) sono suite di utilità di backup complete che sono pensate per backup regolari in rete.
- Regular backups of user data can be realized by a simple script (Sezione 10.2.2).

Gli strumenti base descritti in Sezione 10.1.1 e Sezione 10.1.2 possono essere usati per facilitare il backup di sistema attraverso script personalizzati. Tali script possono essere migliorati con gli strumenti seguenti.

- Il pacchetto `restic` permette backup incrementali (remoti).
- Il pacchetto `rdiff-backup` permette backup incrementali (remoti).
- Il pacchetto `dump` aiuta ad archiviare e ripristinare tutto il file system in maniera incrementale ed efficiente.

Suggerimento

Per imparare ulteriori informazioni sul pacchetto `dump`, vedere i file in `/usr/share/doc/dump/` e ["Is dump really deprecated? \(dump è davvero deprecato?\)"](#).

pacchetto	popcon	dimensione	descrizione
dump	V:1, I:6	352	dump(8) e restore(8) di BSD 4.4 per file system ext2/ext3/ext4
xfsdump	V:0, I:9	854	dump e ripristino con xfsdump(8) e xfsrestore(8) per file system XFS in GNU/Linux e IRIX
backupninja	V:3, I:4	367	sistema di meta-backup leggero ed estensibile
bacula-common	V:10, I:14	2158	Bacula : backup, ripristino e controllo in rete - file comuni di supporto
bacula-client	I:3	183	Bacula : backup, ripristino e controllo in rete - metapacchetto client
bacula-console	V:1, I:4	107	Bacula : backup, ripristino e controllo in rete - console testuale
bacula-server	I:1	183	Bacula : backup, ripristino e controllo in rete - metapacchetto server
amanda-common	V:1, I:2	10030	Amanda : Advanced Maryland Automatic Network Disk Archiver (Librerie)
amanda-client	V:1, I:2	1088	Amanda : Advanced Maryland Automatic Network Disk Archiver (Client)
amanda-server	V:0, I:0	1075	Amanda : Advanced Maryland Automatic Network Disk Archiver (Server)
backup-manager	V:1, I:1	571	strumento di backup a riga di comando
backup2l	V:0, I:1	114	strumento di backup/ripristino per supporti montabile (basato su dischi) che richiede bassa manutenzione
backuppc	V:3, I:3	3183	BackupPC è un sistema ad alte prestazioni di qualità professionale per il backup di PC (basato su dischi)
duplicity	V:9, I:19	1834	backup incrementali (remoti)
flexbackup	V:0, I:0	243	backup incrementali (remoti)
rdiff-backup	V:6, I:14	733	backup incrementali (remoti)
restic	V:1, I:4	22540	backup incrementali (remoti)
slbackup	V:0, I:0	151	backup incrementali (remoti)

Tabella 10.5: Elenco di suite con utilità di backup

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`.

Suggerimento

Si possono ripristinare i dati di configurazione di `debconf` con `"debconf-set-selections debconf-selections"` ed i dati delle selezioni di `dpkg` con `"dpkg --set-selection <dpkg-selections.list"`.

10.3 Infrastruttura di sicurezza dei dati

L'infrastruttura di sicurezza dei dati viene fornita dalla combinazione di strumenti di cifratura dei dati, strumenti message digest e strumenti di firma.

See Sezione 9.9 on `dm-crypt` and `ecryptfs` which implement automatic data encryption infrastructure via Linux kernel modules.

10.3.1 Gestione delle chiavi per GnuPG

Quelli che seguono sono alcuni comandi per [GNU Privacy Guard](#) per la gestione base delle chiavi.

pacchetto	popcon	dimensione	comando	descrizione
gnupg	V:498, I:942	864	gpg(1)	GNU Privacy Guard - strumento OpenPGP di cifratura e firma
gpgv	V:890, I:999	882	gpgv(1)	GNU Privacy Guard - strumento di verifica delle firme
paperkey	V:1, I:13	58	paperkey(1)	estrae solamente le informazioni segrete da chiavi OpenPGP segrete
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)	calcola e controlla message digest MD5
coreutils	V:901, I:999	17478	sha1sum(1)	calcola e controlla message digest SHA1
openssl	V:814, I:994	1465	openssl(1ssl)	calcola message digest con "openssl dgst" (OpenSSL)

Tabella 10.6: Elenco di strumenti per l'infrastruttura di sicurezza dei dati

comando	descrizione
gpg --gen-key	Genera una nuova chiave
gpg --gen-revoke mio_ID_utente	genera una chiave di revoca per mio_ID_utente
gpg --edit-key user_ID	modifica la chiave in modo interattivo, "help" per l'aiuto
gpg -o file --export	esporta tutte le chiavi in un file
gpg --import file	importa tutte le chiavi da un file
gpg --send-keys ID_utente	invia la chiave di ID_utente al server di chiavi
gpg --recv-keys ID_utente	riceve la chiave di ID_utente dal server di chiavi
gpg --list-keys ID_utente	elenca le chiavi di ID_utente
gpg --list-sigs ID_utente	elenca le firme di ID_utente
gpg --check-sigs ID_utente	controlla le firme di ID_utente
gpg --fingerprint ID_utente	controlla le impronte digitali di ID_utente
gpg --refresh-keys	aggiorna il portachiavi locale

Tabella 10.7: Elenco di comandi per GNU Privacy Guard per la gestione delle chiavi

Quelli seguenti sono i significati dei codici di fiducia.

codice	descrizione della fiducia
-	nessuna fiducia assegnata dal proprietario / ancora non calcolata
e	calcolo della fiducia fallito
q	informazioni insufficienti per il calcolo
n	non fidarsi mai di questa chiave
m	marginalmente affidabile
f	completamente fidata
u	definitivamente fidata

Tabella 10.8: Elenco dei significati dei codici di fiducia

Il comando seguente carica la mia chiave "1DD8D791" sul popolare server di chiavi "hkp://keys.gnupg.net".

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

Un buon server di chiavi predefinito impostato in "~/.gnupg/gpg.conf" (o nella vecchia posizione "~/.gnupg/options") si ottiene la voce seguente.

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

Il comando seguente recupera le chiavi sconosciute dal server di chiavi.

```
$ gpg --list-sigs --with-colons | grep '^sig.*\[User ID not found\]' |\
cut -d ':' -f 5 | sort | uniq | xargs gpg --recv-keys
```

In [OpenPGP Public Key Server](#) (versioni pre-0.9.6) esisteva un bug che corrompeva le chiavi con più di 2 sottochiavi. I pacchetti GnuPG più recenti (>1.2.1-2) possono gestire queste sottochiavi corrotte. Vedere l'opzione "--repair-pks-subkey-bug" in `gpg(1)`.

10.3.2 Usare GnuPG su file

Quelli seguenti sono esempi di comandi per usare [GNU Privacy Guard](#) su file.

10.3.3 Usare GnuPG con Mutt

Aggiungere quanto segue al file "~/.muttrc" per evitare che il lento GnuPG venga avviato automaticamente, permettendo allo stesso tempo di richiamarlo digitando "S" nel menu della vista indice.

```
macro index S ":toggle pgp_verify_sig\n"
set pgp_verify_sig=no
```

10.3.4 Usare GnuPG con 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
```

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

comando	descrizione
<code>gpg -a -s file</code>	firma un file in file.asc con corazza ASCII
<code>gpg --armor --sign file</code>	” ”
<code>gpg --clearsign file</code>	inserisce una firma leggibile nel messaggio
<code>gpg --clearsign file mail pippo@example.org</code>	invia un messaggio di posta firmato leggibile a <code>pippo@example.org</code>
<code>gpg --clearsign --not-dash-escaped patchfile</code>	inserisce una firma leggibile in patchfile
<code>gpg --verify file</code>	verifica un file con firma leggibile
<code>gpg -o file.sig -b file</code>	crea una firma staccata
<code>gpg -o file.sig --detach-sig file</code>	” ”
<code>gpg --verify file.sig file</code>	verifica file con file.sig
<code>gpg -o cifr_file.gpg -r nome -e file</code>	cifratura di file nel file binario cifr_file.gpg usando la chiave pubblica indirizzata a nome
<code>gpg -o cifr_file.gpg --recipient nome --encrypt file</code>	” ”
<code>gpg -o cifr_file.asc -a -r nome -e file</code>	cifratura di file nel file con cifratura corazzata ASCII cifr_file.asc usando la chiave pubblica indirizzata a nome
<code>gpg -o cifr_file.gpg -c file</code>	cifratura simmetrica da file a cifr_file.gpg
<code>gpg -o cifr_file.gpg --symmetric file</code>	” ”
<code>gpg -o cifr_file.asc -a -c file</code>	cifratura simmetrica pensata per nome di file nel file con cifratura corazzata ASCII cifr_file.asc
<code>gpg -o file -d cifr_file.gpg -r nome</code>	decifratura
<code>gpg -o file --decrypt cifr_file.gpg</code>	” ”

Tabella 10.9: Elenco di comandi GNU Privacy Guard per file

10.3.5 Somme di controllo MD5

`md5sum(1)` fornisce un'utilità per creare un file digest usando il metodo descritto nella [rfc1321](#) e per verificare i file con esso.

```
$ md5sum foo bar >baz.md5
$ cat baz.md5
d3b07384d113edec49eaa6238ad5ff00  foo
c157a79031e1c40f85931829bc5fc552  bar
$ md5sum -c baz.md5
foo: OK
bar: OK
```

Nota

Il calcolo delle somme di controllo **MD5** è meno dispendioso in termini di CPU di quello delle firme crittografiche di **GNU Privacy Guard (GnuPG)**. Di solito solamente il file digest di più alto livello è firmato crittograficamente per assicurare l'integrità dei dati.

10.4 Strumenti per la fusione di codice sorgente

Esistono molti strumenti per la fusione di codice sorgente. Quello che segue è un elenco di strumenti che hanno catturato la mia attenzione.

10.4.1 Estrarre differenze da file sorgenti

Si possono estrarre le differenze tra due file sorgenti e creare file diff unificati "file.patch0" o "file.patch1", a seconda della posizione del file, con le procedure seguenti.

```
$ diff -u file.old file.new > file.patch0
$ diff -u old/file new/file > file.patch1
```

10.4.2 Fondere aggiornamenti per file sorgenti

I file diff (chiamati anche file patch) sono usati per inviare aggiornamenti per un programma. Chi li riceve applica questo aggiornamento ad un altro file nel modo seguente.

```
$ patch -p0 file < file.patch0
$ patch -p1 file < file.patch1
```

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
```

pacchetto	popcon	dimensione	comando	descrizione
patch	V:123, I:721	248	patch(1)	applica un file diff ad un originale
vim	V:102, I:404	3286	vimdiff(1)	confronta 2 file uno di fianco all'altro in vim
imediff	V:0, I:0	170	imediff(1)	strumento interattivo a tutto schermo per unione di modifiche a 2/3 vie
meld	V:14, I:38	3065	meld(1)	confronta e fonde file (GTK)
wiggle	V:0, I:0	174	wiggle(1)	applica le patch respinte
diffutils	V:883, I:993	1598	diff(1)	confronta i file riga per riga
diffutils	V:883, I:993	1598	diff3(1)	confronta e fonde tre file riga per riga
quilt	V:3, I:32	788	quilt(1)	gestisce serie di patch
wdiff	V:8, I:69	644	wdiff(1)	mostra le differenze di parole tra file di testo
diffstat	V:14, I:146	81	diffstat(1)	produce un istogramma delle modifiche apportate da un diff
patchutils	V:16, I:143	232	combinediff(1)	fa una patch cumulativa da due patch incrementali
patchutils	V:16, I:143	232	dehtmldiff(1)	estrae un diff da una pagina HTML
patchutils	V:16, I:143	232	filterdiff(1)	estrae o esclude diff da un file diff
patchutils	V:16, I:143	232	fixcvsdiff(1)	aggiusta file diff creati da CVS che sono male interpretati da patch(1)
patchutils	V:16, I:143	232	flipdiff(1)	scambia l'ordine di due patch
patchutils	V:16, I:143	232	grepdiff(1)	mostra quali file siano modificati da una patch che fa corrispondenza con un'espressione regolare
patchutils	V:16, I:143	232	interdiff(1)	mostra le differenze tra due file diff unificati
patchutils	V:16, I:143	232	lsdiff(1)	mostra quali file vengano modificati da una patch
patchutils	V:16, I:143	232	recountdiff(1)	calcola conteggi e offset in diff unificati
patchutils	V:16, I:143	232	rediff(1)	aggiusta conteggi ed offset di un diff modificato a mano
patchutils	V:16, I:143	232	splitdiff(1)	separa due patch incrementali
patchutils	V:16, I:143	232	unwrapdiff(1)	ripristina patch il cui contenuto è stato mandato a capo automaticamente
dirdiff	V:0, I:2	166	dirdiff(1)	mostra le differenze ed apporta i cambiamenti tra alberi di directory
docdiff	V:0, I:0	555	docdiff(1)	confronta due file parola per parola / carattere per carattere
makepatch	V:0, I:0	102	makepatch(1)	genera file patch estesi
makepatch	V:0, I:0	102	applypatch(1)	applica file patch estesi

Tabella 10.10: Elenco di strumenti per la fusione di codice sorgente

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.

Debian fornisce servizi Git liberi attraverso il [servizio Debian Salsa](#). La sua documentazione è reperibile su <https://wiki.debian.org/Salsa>.

Here are some Git related packages.

pacchetto	popcon	dimensione	comando	descrizione
git	V:312, I:494	36515	git(7)	Git, il sistema di controllo delle revisioni veloce, scalabile e distribuito
gitk	V:5, I:41	1749	gitk(1)	browser degli archivi Git con interfaccia utente grafica e cronologia
git-gui	V:1, I:23	2343	git-gui(1)	interfaccia utente grafica per Git (senza cronologia)
git-email	V:0, I:11	993	git-send-email(1)	invia una raccolta di patch come messaggio di posta da Git
git-buildpackage	V:1, I:11	4187	git-buildpackage(1)	tra la creazione di pacchetti Debian con 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 sopra a git (Python)
git-doc	I:14	12063	N/D	documentazione ufficiale per Git
gitmagic	I:1	721	N/D	"Git Magic", una guida per Git più semplice da capire

Tabella 10.11: Elenco di pacchetti e comandi relativi a Git

10.5.1 Configurazione del client Git

È possibile che si desideri impostare diverse configurazioni globali, come il nome e l'indirizzo di posta elettronica usati da Git, in `~/.gitconfig` nel modo seguente.

```
$ 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 si è abituati ai comandi di CVS o Subversion, si potrebbe volere impostare alcuni alias per i comandi nel modo seguente.

```
$ git config --global alias.ci "commit -a"
$ git config --global alias.co checkout
```

La configurazione globale può essere controllata con il comando seguente.

```
$ 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.

Git command	funzione
<code>git init</code>	creazione dell'archivio (locale)
<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 <code>treeish1</code> commit and <code>treeish2</code> commit
<code>gitk</code>	GUI display of VCS repository branch history tree

Tabella 10.12: Main Git commands

10.5.3 Git tips

Here are some Git tips.



avvertimento

Non usare stringhe per etichette contenenti spazi, anche se alcuni strumenti come `gitk(1)` lo permettono. Altri comandi `git` potrebbero avere problemi con esse.

Git command line	funzione
<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 loosing 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.)

Tabella 10.13: Git tips

**Attenzione**

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 acceptable for `main` branch but may be acceptable for a topic branch before merging to `main` branch.

**Attenzione**

A partire dall'inizio del 2006, l'invocazione diretta, dalla riga di comando, di un sottocomando di `git` come `"git-xyz"` è diventata deprecata.

Suggerimento

If there is an 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 Documenti di consultazione per Git

Vedere la documentazione seguente.

- [pagina man: git\(1\)](/usr/share/doc/git-doc/git.html) (/usr/share/doc/git-doc/git.html)
- [Manuale utente di Git](/usr/share/doc/git-doc/user-manual.html) (/usr/share/doc/git-doc/user-manual.html)
- [Un tutorial introduttivo su git](/usr/share/doc/git-doc/gittutorial.html) (/usr/share/doc/git-doc/gittutorial.html)
- [Un tutorial introduttivo su git: parte seconda](/usr/share/doc/git-doc/gittutorial-2.html) (/usr/share/doc/git-doc/gittutorial-2.html)
- [Everyday GIT With 20 Commands Or So](/usr/share/doc/git-doc/giteveryday.html) (/usr/share/doc/git-doc/giteveryday.html)
- [Git Magic](/usr/share/doc/gitmagic/html/index.html) (/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.

pacchetto	popcon	dimensione	strumento	tipo di VCS	commento
mercurial	V:6, I:44	1053	Mercurial	distribuito	DVCS in Python e un po' di C
darcs	V:0, I:7	23159	Darcs	distribuito	DVCS con algebra intelligente per le patch (lento)
bzd	V:1, I:15	28	Bazaar	distribuito	DVCS influenced by t la written in Python (historic)
tla	V:0, I:2	1011	GNU arch	distribuito	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	visualizzazione GUI di alberi di archivi VCS (CVS, Subversion, RCS)
rcs	V:3, I:19	562	RCS	locale	" Unix SCCS done right" (historic)
cssc	V:0, I:2	2044	CSSC	locale	clone of the Unix SCCS (historic)

Tabella 10.14: List of other version control system tools

Capitolo 11

Conversione di dati

In questo capitolo sono descritti strumenti e trucchi per convertire tra diversi formati di dati in un sistema Debian. Gli strumenti standard sono eccellenti, ma il supporto per i formati proprietari per i dati è limitato.

11.1 Strumenti di conversione di dati testuali

Quelli che seguono sono pacchetti per la conversione dei dati testuali che hanno attirato la mia attenzione.

pacchetto	popcon	dimensione	parola chiave	descrizione
libc6	V:946, I:999	12772	set di caratteri	convertitore della codifica del testo tra localizzazioni usando <code>iconv(1)</code> (fondamentale)
recode	V:3, I:25	603	set caratteri+carattere fine riga	convertitore della codifica del testo tra localizzazioni (versatile, maggior numero di alias e funzionalità)
konwert	V:1, I:52	134	set di caratteri	convertitore della codifica tra localizzazioni (elegante)
nkf	V:0, I:11	358	set di caratteri	traduttore del set di caratteri per il giapponese
tcs	V:0, I:0	518	set di caratteri	traduttore del set di caratteri
unaccent	V:0, I:0	29	set di caratteri	sostituisce le lettere accentate con le equivalenti senza accento
tofrodos	V:1, I:23	55	carattere fine riga	convertitore del formato di testo tra DOS e Unix: <code>fromdos(1)</code> e <code>todos(1)</code>
macutils	V:0, I:0	298	carattere fine riga	convertitore del formato di testo tra Macintosh e Unix: <code>frommac(1)</code> e <code>tomac(1)</code>

Tabella 11.1: Elenco di strumenti di conversione di dati testuali

11.1.1 Convertire un file di testo con `iconv`

Suggerimento

`iconv(1)` viene fornito come parte del pacchetto `libc6` ed è sempre disponibile praticamente su tutti i sistemi in stile Unix, per fare la conversione della codifica dei caratteri.

Si può convertire la codifica di un file di testo con `iconv(1)` nel modo seguente.

```
$ iconv -f encoding1 -t encoding2 input.txt >output.txt
```

Nel trovare corrispondenze con i valori delle codifiche non viene tenuto conto delle lettere maiuscole o minuscole e vengono ignorati i caratteri ”-” e ”_”. Si possono controllare le codifiche supportate con il comando ”`iconv -l`”.

valore della codifica	uso
ASCII	American Standard Code for Information Interchange , codifica americana standard per lo scambio di informazioni, codice a 7 bit senza caratteri accentati
UTF-8	attuale standard multilingua per tutti i sistemi operativi moderni
ISO-8859-1	vecchio standard per le lingue europee occidentali, ASCII + caratteri accentati
ISO-8859-2	vecchio standard per le lingue europee orientali, ASCII + caratteri accentati
ISO-8859-15	vecchio standard per le lingue europee occidentali, ISO-8859-1 con simbolo dell'euro
CP850	code page 850, caratteri Microsoft DOS con caratteri grafici per le lingue europee occidentali, variante di ISO-8859-1
CP932	code page 932, variante in stile Microsoft Windows di Shift-JIS per il giapponese
CP936	code page 936, variante in stile Microsoft Windows di GB2312 , GBK o GB18030 per il cinese semplificato
CP949	code page 949, variante in stile Microsoft Windows di EUC-KR o Unified Hangul Code per il coreano
CP950	code page 950, variante in stile Microsoft Windows di Big5 per il cinese tradizionale
CP1251	code page 1251, codifica in stile Microsoft Windows per l'alfabeto cirillico
CP1252	code page 1252, variante in stile Microsoft Windows di ISO-8859-15 per le lingue europee occidentali
KOI8-R	vecchio standard UNIX russo per l'alfabeto cirillico
ISO-2022-JP	codifica standard per la posta elettronica in giapponese che usa solo codici a 7 bit
eucJP	vecchio standard UNIX giapponese con codici a 8 bit completamente diverso da Shift-JIS
Shift-JIS	standard JIS X 0208 Appendix 1 per il giapponese (vedere CP932)

Tabella 11.2: Elenco dei valori delle codifiche e loro uso

Nota

Some encodings are only supported for the data conversion and are not used as locale values (Sezione [8.1](#)).

Per i set di caratteri contenuti in un singolo byte, come i set di caratteri [ASCII](#) e [ISO-8859](#), la [codifica di carattere](#) è quasi la stessa cosa del set di caratteri.

Per i set di caratteri con molti caratteri, come [JIS X 0213](#) per il giapponese o [Universal Character Set \(UCS, Unicode, ISO-10646-1\)](#) per praticamente tutte le lingue, esistono molti schemi di codifica per inserirli nella sequenza dei byte dati.

- [EUC](#) e [ISO/IEC 2022](#) (chiamato anche [JIS X 0202](#)) per il giapponese
- [UTF-8](#), [UTF-16/UCS-2](#) e [UTF-32/UCS-4](#) per Unicode

In questi casi c'è una netta differenza tra il set di caratteri e la codifica di caratteri.

Il termine [code page](#) è usato come sinonimo di tabella di codifica dei caratteri per alcune tabelle specifiche di produttori.

Nota

Notare che la maggior parte dei sistemi di codifica condividono con ASCII gli stessi codici per i caratteri a 7 bit. Ci sono però alcune eccezioni. Se si stanno convertendo dati di vecchi programmi C o URL in giapponese dal formato di codifica volgarmente chiamato Shift-JIS in formato UTF-8, usare "CP932" come nome di codifica invece di "shift-JIS" per ottenere i risultati attesi: 0x5C → "\"" e 0x7E → "~". Altrimenti questi vengono convertiti nei caratteri sbagliati.

Suggerimento

Si può anche usare `recode(1)` che offre più della semplice combinazione delle funzionalità di `iconv(1)`, `fromdos(1)`, `todos(1)`, `frommac(1)` e `tomac(1)`. Per maggiori informazioni vedere "info recode".

11.1.2 Controllare se un file è in UTF-8 con `iconv`

Si può controllare se un file di testo è nella codifica UTF-8 usando `iconv(1)` nel modo seguente.

```
$ iconv -f utf8 -t utf8 input.txt >/dev/null || echo "non-UTF-8 found"
```

Suggerimento

Usare l'opzione "--verbose" nell'esempio precedente per trovare il primo carattere non UTF-8.

11.1.3 Convertire nomi di file con `iconv`

Ecco uno script d'esempio per convertire in una directory la codifica dei nomi di file dai nomi creati in un vecchio sistema operativo a quelli UTF-8 moderni.

```
#!/bin/sh
ENCDN=iso-8859-1
for x in *;
do
mv "$x" "$(echo "$x" | iconv -f $ENCDN -t utf-8)"
done
```

La variabile "\$ENCDN" specifica la codifica originale usata per i nomi di file nei sistemi operativi più vecchi come in Tabella 11.2.

Per scenari più complessi, montare, usando come opzione di `mount(8)` la codifica appropriata (vedere Sezione 8.1.3), il file system (ad esempio una partizione in un disco fisso) contenente tali nomi di file e copiare usando il comando "cp -a" il suo intero contenuto in un altro file system montato come UTF-8.

11.1.4 Conversione del carattere di fine riga

Il formato dei file di testo e specificatamente il codice EOL (End of line, fine riga) è dipendente dalla piattaforma.

I programmi di conversione del formato di EOL, `fromdos(1)`, `todos(1)`, `frommac(1)` e `tomac(1)`, sono piuttosto comodi. Anche `recode(1)` è utile.

Nota

Alcuni dati nel sistema Debian, come i dati della pagina wiki per il pacchetto `python-moinmoin`, usano CR-LF in stile MSDOS come codice EOL. Perciò la regola sopra descritta è solamente una regola generale.

piattaforma	codice EOL	carattere di controllo	decimale	esadecimale
Debian (unix)	LF	^J	10	0A
MSDOS e Windows	CR-LF	^M^J	13 10	0D 0A
Macintosh di Apple	CR	^M	13	0D

Tabella 11.3: Elenco di stili per EOL per differenti piattaforme

Nota

La maggior parte degli editor (es. `vim`, `emacs`, `gedit`, ...) può gestire file con EOL in stile MSDOS in modo trasparente.

Suggerimento

È meglio usare `sed -e '/\r$/!s/$/\r/'` invece di `todos(1)` quando si desidera unificare lo stile EOL allo stile MSDOS da uno stile misto MSDOS e Unix (ad esempio dopo la fusione di 2 file in stile MSDOS con `diff3(1)`). Questo perché `todos` aggiunge CR a tutte le righe.

11.1.5 Conversione di tabulazioni

Esistono alcuni programmi popolari specializzati nella conversione dei codici di tabulazione.

funzione	bsdmainutils	coreutils
espande le tabulazioni in spazi	<code>"col -x"</code>	<code>expand</code>
de-espande gli spazi in tabulazioni	<code>"col -h"</code>	<code>unexpand</code>

Tabella 11.4: Elenco di comandi di conversione di TAB dai pacchetti `bsdmainutils` e `coreutils`

`indent(1)` nel pacchetto `indent` riformatta completamente gli spazi bianchi nei programmi C.

Anche programmi editor come `vim` ed `emacs` possono essere usati per la conversione di TAB. Per esempio con `vim`, è possibile espandere i TAB con la sequenza di comandi `":set expandtab"` e `":%retab"`. Si può annullare questa azione con la sequenza di comandi `":set noexpandtab"` e `":%retab!"`.

11.1.6 Editor con auto-conversione

Gli editor moderni intelligenti come il programma `vim` sono piuttosto bravi e gestiscono bene qualsiasi sistema di codifica e formato di file. Per una migliore compatibilità questi editor andrebbero usati con la localizzazione UTF-8 in console con capacità UTF-8.

Un vecchio file di testo Unix in lingua europea occidentale, `"u-file.txt"` salvato nella codifica `latin1 (iso-8859-1)` può essere modificato con `vim` semplicemente con il comando seguente.

```
$ vim u-file.txt
```

Ciò è reso possibile dal fatto che il meccanismo di rivelazione automatica della codifica dei file di `vim` presuppone come prima cosa che la codifica sia UTF-8 e, se ciò fallisce, presuppone che sia `latin1`.

Un vecchio file di testo Unix in lingua polacca, `"pu-file.txt"` salvato nella codifica `latin1 (iso-8859-1)` può essere modificato con `vim` con il comando seguente.

```
$ vim '+e ++enc=latin2 pu-file.txt'
```

Un vecchio file di testo Unix in giapponese, "ju-file.txt" salvato nella codifica eucJP può essere modificato con vim con il comando seguente.

```
$ vim '+e ++enc=eucJP ju-file.txt'
```

Un vecchio file di testo MS-Windows in giapponese, "jw-file.txt" salvato nella cosiddetta codifica shift-JIS (più precisamente CP932) può essere modificato con vim con il comando seguente.

```
$ vim '+e ++enc=CP932 ++ff=dos jw-file.txt'
```

Quando un file viene aperto con le opzioni "++enc" e "++ff", l'uso di ":w" nella riga di comando di Vim lo salva nel formato originale sovrascrivendo il file originale. Si può anche specificare nella riga di comando di Vim il formato e il nome con cui salvare il file, ad esempio ":w ++enc=utf8 nuovo.txt".

Fare riferimento a mbyte.txt per il "supporto di testi multi-byte" nell'aiuto in linea di vim e a Tabella 11.2 per i valori delle localizzazione usati con "++enc".

La famiglia di programmi emacs può svolgere funzioni equivalenti.

11.1.7 Estrazione del testo puro

Il comando seguente legge una pagina web mettendola in un file di testo. È molto utile quando si copiano configurazioni dal Web o per applicare strumenti di testo base Unix come grep(1) sulla pagina web.

```
$ w3m -dump http://www.remote-site.com/help-info.html >textfile
```

In modo analogo, si possono estrarre dati in testo puro da altri formati con gli strumenti seguenti.

pacchetto	popcon	dimensione	parola chiave	funzione
w3m	V:25, I:252	2367	html → testo puro	conversione da HTML a testo semplice con il comando "w3m -dump"
html2text	V:2, I:29	274	html → testo puro	convertitore avanzato da HTML a testo semplice (ISO 8859-1)
lynx	V:11, I:97	1948	html → testo puro	conversione da HTML a testo semplice con il comando "lynx -dump"
elinks	V:5, I:27	1767	html → testo puro	conversione da HTML a testo semplice con il comando "elinks -dump"
links	V:5, I:38	2250	html → testo puro	conversione da HTML a testo semplice con il comando "links -dump"
links2	V:1, I:14	5418	html → testo puro	conversione da HTML a testo semplice con il comando "links2 -dump"
catdoc	V:22, I:125	675	MSWord → testo puro, TeX	converte file MSWord in testo puro o TeX
antiword	V:1, I:10	589	MSWord → testo puro, ps	converte file MSWord in testo puro o ps
pstotext	V:0, I:2	126	ps/pdf → testo puro	estrae testo da file PostScript e PDF
unhtml	V:0, I:0	43	html → testo puro	rimuove i tag marcatori da file HTML
odt2txt	V:1, I:7	60	odt → testo puro	converte da OpenDocument Text in testo puro

Tabella 11.5: Elenco di strumenti per estrarre dati in testo puro

11.1.8 Evidenziare e formattare dati in puro testo

È possibile evidenziare e formattare dati in testo puro usando gli strumenti seguenti.

pacchetto	popcon	dimensione	parola chiave	descrizione
vim-runtime	V:18, I:439	32134	evidenziare	MACRO Vim per convertire codice sorgente in HTML con <code>":source \$VIMRUNTIME/syntax/html.vim"</code>
cxref	V:0, I:0	1193	c → html	convertitore per programmi C in latex e HTML (linguaggio C)
src2tex	V:0, I:0	622	evidenziare	converte molti codici sorgenti in TeX (linguaggio C)
source-highlight	V:0, I:7	1992	evidenziare	converte molti codici sorgenti in file HTML, XHTML, LaTeX, Texinfo, sequenze di escape per colori ANSI e DocBook con evidenziazione (C++)
highlight	V:1, I:10	1083	evidenziare	converte molti codici sorgenti in file HTML, XHTML, RTF, LaTeX, TeX o XSL-FO con evidenziazione (C++)
grc	V:0, I:3	208	testo → colori	colorazione generica per tutto (Python)
python-docutils	V:9, I:112	1771	testo → tutto	formattatore di documenti ReStructured Text in XML (Python)
pandoc	V:8, I:48	151714	testo → tutto	convertitore universale di marcatori (Haskel)
markdown	V:0, I:8	57	testo → html	formattatore markdown di documenti di testo in (X)HTML (Perl)
txt2html	V:0, I:3	259	testo → html	convertitore da testo semplice in HTML (Perl)
asciidoc	I:12	81	testo → tutto	formattatore AsciiDoc di documenti di testo in XML/HTML (Python)

Tabella 11.6: Elenco di strumenti per evidenziare dati in testo puro

11.2 Dati XML

XML (Extensible Markup Language) è un linguaggio a marcatori per documenti contenenti informazioni strutturate.

Vedere informazioni introduttive su [XML.COM](#).

- ["Cosa è XML?"](#)
- ["Cosa è XSLT?"](#)
- ["Cosa è XSL-FO?"](#)
- ["Cosa è XLink?"](#)

11.2.1 Suggerimenti base per XML

I testi XML hanno un aspetto simile all'**HTML**. Permettono di gestire formati di output multipli per un documento. Un facile sistema per XML è il pacchetto `docbook-xsl` che è stato usato per questo documento.

Ogni file XML inizia con una dichiarazione XML standard come la seguente.

```
?xml version="1.0" encoding="UTF-8"?
```

La sintassi di base per un elemento XML usa un marcatore come il seguente.

```
name attribute="value"content/name
```

Gli elementi XML senza contenuto sono indicati nella seguente forma breve.

```
name attribute="value"/
```

La parte "attributo="valore"" negli esempi precedenti è opzionale.

In XML una sezione di commento è marcata nel modo seguente.

```
!-- comment --
```

A parte l'aggiunta dei marcatori, XML richiede solo una minima conversione dei contenuti usando entità predefinite per i caratteri seguenti.

entità predefinita	carattere in cui convertire
"	" : virgolette
'	' : apostrofo
<	< : minore-di
>	> : maggiore-di
&	& : e-commerciale

Tabella 11.7: Elenco di entità predefinite per XML



Attenzione

"<" e "&" non possono essere usati in attributi o elementi.

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

Fintanto che i marcatori XML sono usati in modo coerente con un certo set dei nomi di tag (qualche dato usato come contenuto o valore di attributo), la conversione in un altro XML è un compito banale usando [XSLT \(Extensible Stylesheet Language Transformations\)](#).

11.2.2 Elaborazione XML

Sono disponibili molti strumenti per elaborare file XML, come l'[XSL \(Extensible Stylesheet Language\)](#).

Fondamentalmente, una volta creato un file XML ben formato, lo si può convertire in qualsiasi formato usando [XSLT \(Extensible Stylesheet Language Transformation\)](#).

[XSL-FO \(Extensible Stylesheet Language for Formatting Object\)](#), linguaggio per fogli di stile estensibile per la formattazione di oggetti, è pensato per essere una soluzione per la formattazione. Il pacchetto `fop` è una novità nell'archivio Debian `main` a causa delle sue dipendenze dal [linguaggio di programmazione Java](#). Perciò il codice LaTeX è solitamente generato dall'XML usando XSLT ed il sistema LaTeX viene usato per creare file adatti alla stampa come DVI, PostScript e PDF.

Dato che XML è un sottoinsieme di [SGML \(Standard Generalized Markup Language\)](#), può essere elaborato dagli strumenti completi disponibili per SGML, come [DSSSL \(Document Style Semantics and Specification Language\)](#).

pacchetto	popcon	dimensione	parola chiave	descrizione
docbook-xml	V:14, I:264	2133	xml	DTD (Document Type Definition) XML per DocBook
docbook-xsl	V:12, I:154	14870	xml/xslt	fogli di stile XSL per elaborare XML DocBook con XSLT in vari formati di output
xsltproc	V:14, I:100	160	xslt	elaboratore a riga di comando XSLT (XML → XML, HTML, testo semplice, ecc.)
xmlto	V:1, I:22	130	xml/xslt	convertitore da-XML-a-tutto con XSLT
fop	V:1, I:20	291	xml/xsl-fo	converte file XML DocBook in PDF
dblatex	V:2, I:15	4643	xml/xslt	converte file DocBook con XSLT in documenti DVI, PostScript, PDF
dbtoepub	V:0, I:0	37	xml/xslt	convertitore da XML DocBook a .epub

Tabella 11.8: Elenco di strumenti XML

pacchetto	popcon	dimensione	parola chiave	descrizione
openjade	V:2, I:36	1019	dsssl	elaboratore standard DSSSL ISO/IEC 10179:1996 (più recente)
docbook-dsssl	V:1, I:21	2604	xml/dsssl	fogli di stile DSSSL per elaborare XML DocBook con DSSSL in vari formati di output
docbook-utils	V:0, I:15	281	xml/dsssl	utilità per file DocBook, inclusa la conversione in altri formati (HTML, RTF, PS, man, PDF) con comandi <code>docbook2*</code> con DSSSL
sgml2x	V:0, I:0	90	SGML/dsssl	convertitore da SGML e XML che usa fogli di stile DSSSL

Tabella 11.9: Elenco di strumenti DSSSL

Suggerimento

A volte `ye lp` di [GNOME](#) è utile per leggere file XML [DocBook](#) direttamente dato che li visualizza in modo decente in X.

11.2.3 Estrazione di dati XML

Si possono estrarre dati HTML o XML da altri formati usando gli strumenti seguenti.

pacchetto	popcon	dimensione	parola chiave	descrizione
man2html	V:0, I:2	138	pagine man → html	convertitore da pagine man a HTML (supporto CGI)
doclifter	V:0, I:0	451	troff → xml	convertitore da troff a XML DocBook
texi2html	V:0, I:7	1833	texi → html	convertitore da texinfo a HTML
info2www	V:1, I:3	76	info → html	convertitore da GNU info a HTML (supporto CGI)
wv	V:0, I:6	717	MSWord → tutto	convertitore di documenti da Microsoft Word a HTML, LaTeX, ecc.
unrtf	V:0, I:4	148	rtf → html	convertitore di documenti da RTF a HTML, ecc.
ooo2dbk	V:0, I:0	217	sxw → xml	convertitore da documenti SXW OpenOffice.org in XML DocBook
wp2x	V:0, I:0	202	WordPerfect → tutto	da file WordPerfect 5.0 e 5.1 a TeX, LaTeX, troff, GML e HTML

Tabella 11.10: Elenco di strumenti di estrazione di dati XML

11.2.4 The XML data lint

I file HTML non-XML possono essere convertiti in XHTML che è un'istanza di XML ben strutturato. XHTML può essere elaborato con strumenti XML.

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

pacchetto	popcon	dimensione	funzione	descrizione
libxml2-utils	V:21, I:234	181	xml ↔ html ↔ xhtml	strumento XML a riga di comando con <code>xmllint(1)</code> (controllo di sintassi, riformattazione, eliminazione sporcizia, ...)
tidy	V:1, I:13	84	xml ↔ html ↔ xhtml	controllore della sintassi e riformattatore per HTML
weblint-perl	V:0, I:1	32	lint	strumento di controllo della sintassi e dello stile di base per HTML
linklint	V:0, I:0	344	link check	controllore veloce di collegamenti e strumento per manutenzione di siti web

Tabella 11.11: Elenco di strumenti per belle stampe XML

Una volta che è stato generato codice XML corretto, si può usare la tecnologia XSLT per estrarre dati in base al contesto dei marcatori

11.3 Impaginazione

Per semplici impaginazioni si può usare il programma Unix [troff](#) sviluppato da AT&T. Viene di solito usato per creare pagine man.

TeX, creato da Donald Knuth è uno strumento di impaginazione molto potente ed è lo standard di fatto. **LaTeX**, scritto originariamente da Leslie Lamport, permette un accesso alla potenza di TeX a più alto livello.

pacchetto	popcon	dimensione	parola chiave	descrizione
texlive	V:3, I:48	72	(La)TeX	sistema TeX per impaginazione, anteprima e stampa
groff	V:3, I:59	11870	troff	sistema di formattazione di testi GNU troff

Tabella 11.12: Elenco di strumenti per impaginazione

11.3.1 Impaginazione roff

Tradizionalmente il sistema di elaborazione di testi Unix principale è [roff](#). Vedere [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".

Si può leggere o stampare un buon tutorial e documento di consultazione sulla [macro](#) "me" in `"/usr/share/doc/groff/"`, dopo aver installato il pacchetto `groff`.

Suggerimento

"`groff -Tascii -me -`" produce output in puro testo con [codici di escape ANSI](#). Se si desidera produrre un output in stile pagine man con molti "`^H`" e "`_`", usare invece "`GROFF_NO_SGR=1 groff -Tascii -me -`".

Suggerimento

Per rimuovere i "`^H`" e "`_`" da un file di testo generato con `groff`, filtrarlo con "`col -b -x`".

11.3.2 TeX/LaTeX

La distribuzione software [TeX Live](#) offre un sistema TeX completo. Il metapacchetto `texlive` fornisce una buona selezione dei pacchetti [TeX Live](#) che dovrebbe essere sufficiente per la maggior parte dei compiti più comuni.

Ci sono molti documenti consultabili disponibili per [TeX](#) e [LaTeX](#).

- [reTeX HOWTO: The Linux-teTeX Local Guide](#)
- [tex\(1\)](#)
- [latex\(1\)](#)
- [texdoc\(1\)](#)
- [texdoctk\(1\)](#)
- "The TeXbook", di Donald E. Knuth, (Addison-Wesley)
- "LaTeX - A Document Preparation System", di Leslie Lamport, (Addison-Wesley)
- "The LaTeX Companion", di Goossens, Mittelbach, Samarin, (Addison-Wesley)

Questo è l'ambiente di impaginazione più potente. Molti elaboratori [SGML](#) lo usano come backend per l'elaborazione del testo. [Lyx](#), fornito dal pacchetto `lyx`, e [GNUTeXmacs](#), fornito dal pacchetto `texmacs`, offrono un bell'ambiente [WYSIWYG](#) per [LaTeX](#); inoltre molti usano [Emacs](#) e [Vim](#) come scelta di editor per i sorgenti.

Sono disponibili molte risorse in rete.

- The TEX Live Guide - TEX Live 2007 ("</usr/share/doc/texlive-doc-base/english/texlive-en/live.html>") (pacchetto `texlive-doc-base`)
- [A Simple Guide to Latex/Lyx](#)
- [Word Processing Using LaTeX](#)
- [Local User Guide to teTeX/LaTeX](#)

Quando i documenti crescono di dimensioni, a volte TeX può generare errori. Per risolvere questo problema si deve aumentare la dimensione di pool in "`/etc/texmf/texmf.cnf`" (o in modo più corretto, modificare "`/etc/texmf/texmf.d/95NonPath`" ed eseguire `update-texmf(8)`).

Nota

Il sorgente TeX di "The TeXbook" è disponibile all'indirizzo <http://tug.ctan.org/tex-archive/systems/knuth/dist/tex-texbook.tex>. Questo file contiene la maggior parte delle macro necessarie. Mi è stato detto che si può elaborare questo documento con `tex(1)` dopo aver commentato le righe da 7 a 10 ed aggiungendo "`\input manmac \proofmodefalse`". È caldamente raccomandabile comprare questo libro (e tutti gli altri libri di Donald E. Knuth) invece di usare le versioni in rete, ma il sorgente è un meraviglioso esempio di input TeX!

11.3.3 Fare una bella stampa di una pagina di manuale

Si può stampare una pagina di manuale in PostScript in un bel modo usando uno dei comandi seguenti.

```
$ man -Tps some_manpage | lpr
```

11.3.4 Creare una pagina di manuale

Benché sia possibile scrivere una pagina di manuale (pagina man) in formato `troff` semplice, ci sono alcuni pacchetti di aiuto per farlo.

pacchetto	popcon	dimensione	parola chiave	descrizione
docbook-to-man	V:0, I:12	191	SGML → pagina	convertitore da SGML DocDook in macro roff man
help2man	V:0, I:10	491	testo → pagina	generatore automatico di pagine man da --help
info2man	V:0, I:0	135	info → pagina	convertitore da GNU info a POD o pagine man
txt2man	V:0, I:1	114	testo → pagina	converte testo in puro ASCII nel formato delle pagine man

Tabella 11.13: Elenco di pacchetti che aiutano a creare una pagina man

11.4 Dati stampabili

In un sistema Debian i dati stampabili sono presentati in formato `PostScript`. `CUPS (Common Unix Printing System)` usa `Ghostscript` come suo programma backend per la rasterizzazione per le stampanti non-`PostScript`.

11.4.1 Ghostscript

Il cuore centrale della manipolazione dei dati da stampare è l'interprete `PostScript (PS) Ghostscript` che genera immagini raster.

Suggerimento

La configurazione di `Ghostscript` può essere visualizzata con "`gs -h`".

pacchetto	popcon	dimensione	descrizione
ghostscript	V:256, I:576	231	interprete PostScript/PDF GPL Ghostscript
ghostscript-x	V:12, I:63	223	interprete PostScript/PDF GPL Ghostscript - supporto per display X
libpoppler102	V:55, I:89	4274	libreria di rendering per PDF risultato di un fork del visualizzatore di PDF xpdf
libpoppler-glib8	V:228, I:463	445	libreria di rendering per PDF (libreria condivisa basata su GLib)
poppler-data	V:102, I:621	13090	CMaps per la libreria di rendering per PDF (per supporto CJK: Adobe-*)

Tabella 11.14: Elenco di interpreti PostScript Ghostscript

11.4.2 Unire due file PS o PDF

È possibile unire due file [PostScript \(PS\)](#) o [PDF \(Portable Document Format\)](#) usando `gs(1)` di 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

Il formato [PDF](#) che è un formato ampiamente usato per dati stampabili multiplatforma, è essenzialmente il formato [PS](#) compresso con alcune funzionalità ed estensioni aggiuntive.

Suggerimento

Per manipolare dalla riga di comando documenti PostScript sono utili `psmerge(1)` e altri comandi nel pacchetto `psutils`. Anche `pdftk(1)`, nel pacchetto `pdftk`, è utile per manipolare documenti PDF.

11.4.3 Utilità per dati stampabili

Quello che segue è un elenco di pacchetti con utilità per dati stampabili che hanno attirato la mia attenzione.

11.4.4 Stampare con CUPS

Entrambi i comandi `lp(1)` e `lpr(1)` forniti da [CUPS \(Common Unix Printing System\)](#) forniscono opzioni per stampe personalizzate dei dati stampabili.

Si possono stampare 3 copie fascicolate di un file usando uno dei comandi seguenti.

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

```
$ lpr -#3 -o Collate=True filename
```

Si può personalizzare ulteriormente l'operazione di stampa usando opzioni come `"-o number-up=2"`, `"-o page-set=even"`, `"-o page-set=odd"`, `"-o scaling=200"`, `"-o natural-scaling=200"`, ecc., documentate in [Stampa dalla riga di comando ed opzioni relative](#).

11.5 Conversione dei dati di posta

Quelli che seguono sono pacchetti per la conversione dei dati di posta che hanno attirato la mia attenzione.

pacchetto	popcon	dimensione	parola chiave	descrizione
poppler-utils	V:245, I:413	689	pdf → ps, testo, ...	utilità PDF: pdftops , pdfinfo , pdfimages , pdftotext , pdffonts
psutils	V:6, I:98	219	ps → ps	strumenti di conversione di documenti PostScript
poster	V:0, I:4	58	ps → ps	crea grandi poster da pagine PostScript
enscript	V:1, I:19	2132	testo → ps, html, rtf	converte testo ASCII in PostScript, HTML, RTF o Pretty-Print
a2ps	V:0, I:14	3648	testo → ps	convertitore e creatore di belle stampe "Da tutto a PostScript"
pdftk	I:50	28	pdf → pdf	strumento di conversione di documenti PDF: pdftk
html2ps	V:0, I:3	249	html → ps	converte da HTML a PostScript
gnuhtml2latex	V:0, I:1	27	html → latex	convertitore da HTML a latex
latex2rtf	V:0, I:6	480	latex → rtf	converte documenti da LaTeX a RTF leggibili da MS Word
ps2eps	V:3, I:62	99	ps → eps	converte da PostScript a EPS (Encapsulated PostScript)
e2ps	V:0, I:0	109	testo → ps	convertitore da testo a PostScript con supporto per la codifica giapponese
impose+	V:0, I:0	119	ps → ps	Utilità PostScript
trueprint	V:0, I:0	146	testo → ps	belle stampe di molti tipi di codice sorgente (C, C++, Java, Pascal, Perl, Pike, Sh e Verilog) in PostScript. (linguaggio C)
pdf2svg	V:0, I:4	30	ps → svg	convertitore da PDF al formato SVG (Scalable Vector Graphics)
pdf2oipe	V:0, I:0	71	ps → ipe	convertitore da PDF al formato XML di IPE

Tabella 11.15: Elenco di utilità per dati stampabili

pacchetto	popcon	dimensione	parola chiave	descrizione
sharutils	V:4, I:52	1421	posta	shar(1) , unshar(1) , uuencode(1) , uudecode(1)
mpack	V:1, I:17	106	MIME	codifica e decodifica di messaggi MIME : mpack(1) e munpack(1)
tnef	V:1, I:10	110	ms-tnef	spacchettamento di allegati MIME di tipo "application/ms-tnef" che è un formato esclusivo di Microsoft
uudeview	V:0, I:4	109	posta	codificatore e decodificatore per i seguenti formati: uuencode , xxencode , BASE64 , quoted printable e BinHex

Tabella 11.16: Elenco di pacchetto che aiutano a convertire dati di posta

Suggerimento

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.

11.5.1 Nozioni di base sui dati di posta

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 [Tabella 11.2](#)).

Il formato standard per l'archiviazione della posta è mbox strutturato seguendo la [RFC2822 \(che aggiorna RFC822\)](#). Vedere [mbox\(5\)](#) (fornito dal pacchetto `mutt`).

Per le lingue europee, per la posta viene di solito usato "Content-Transfer-Encoding: quoted-printable" con il set di caratteri ISO-8859-1 dato che non ci sono molti caratteri a 8 bit. Se il testo europeo è codificato in UTF-8, è probabile venga usato "Content-Transfer-Encoding: quoted-printable" dato che sono per la maggior parte dati a 7 bit.

Per il giapponese, per la posta viene tradizionalmente usato "Content-Type: text/plain; charset=ISO-2022-JP" per mantenere il testo a 7 bit. Ma sistemi Microsoft più vecchi possono inviare dati di posta in Shift-JIS senza una dichiarazione appropriata. Se il testo giapponese è codificato in UTF-8, è probabile venga usato [Base64](#) dato che contiene molti dati a 8 bit. La situazione delle altre lingue asiatiche è simile.

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 si usano altri formati di archiviazione, spostarli nel formato mbox è un buon primo passo. Un versatile programma client come `mutt(1)` può essere di aiuto in questo caso.

Si può suddividere il contenuto di una casella di posta nei singoli messaggi usando `procmail(1)` e `formail(1)`.

Ciascun messaggio di posta può essere spackettato usando `munpack(1)`, dal pacchetto `mpack`, (o con altri strumenti specializzati) per ottenere i contenuti codificati MIME.

11.6 Strumenti per dati grafici

Quelli che seguono sono pacchetti per la conversione, la modifica e l'organizzazione di dati grafici che hanno attirato la mia attenzione.

Suggerimento

Cercare ulteriori strumenti per le immagini usando l'espressione regolare `"~Gworks-with::image"` in `aptitude(8)` (vedere [Sezione 2.2.6](#)).

Sebbene programmi con interfaccia utente grafica come `gimp(1)` siano molto potenti, strumenti a riga di comando come `imagemagick(1)` sono piuttosto utili per manipolare in modo automatico immagini usando script.

Il formato standard di fatto per i file immagini delle fotocamere digitali è [EXIF \(Exchangeable Image File Format\)](#) che è il formato per file immagine [JPEG](#) con tag aggiuntivi con metadati. Può contenere informazioni come la data, l'orario e le impostazioni della fotocamera.

I diritti sulla [compressione senza perdita di dati Lempel-Ziv-Welch \(LZW\)](#) sono scaduti. Le utilità [GIF \(Graphics Interchange Format\)](#) che usano il metodo di compressione LZW sono ora disponibili liberamente sul sistema Debian.

pacchetto	popcon	dimensione	parola chiave	descrizione
gimp	V:58, I:315	19706	immagine(bitmap)	GNU Image Manipulation Program, programma GNU di manipolazione di immagini
imagemagick	I:373	220	immagine(bitmap)	programmi di manipolazione immagini
graphicsmagick	V:2, I:17	5242	immagine(bitmap)	programmi di manipolazione di immagini (fork di Imagemagick)
xsane	V:14, I:165	2346	immagine(bitmap)	frontend X11 basato su GTK per SANE (Scanner Access Now Easy)
netpbm	V:29, I:382	5056	immagine(bitmap)	strumenti di conversione di dati grafici
icoutils	V:18, I:115	221	png ↔ ico(bitmap)	converte icone e puntatori MS Windows da e verso il formato PNG (favicon.ico)
scribus	V:1, I:22	30370	ps/pdf/SVG/...	editor DTP Scribus
libreoffice-draw	V:150, I:416	14596	immagine(vettoriale)	draw per ufficio LibreOffice - disegno
inkscape	V:42, I:188	84892	immagine(vettoriale)	editor SVG (Scalable Vector Graphics)
dia	V:4, I:30	3727	immagine(vettoriale)	editor di diagrammi (Gtk)
xfig	V:2, I:14	1793	immagine(vettoriale)	strumento per la generazione interattiva di figure in X11
pstoedit	V:3, I:87	988	ps/pdf → immagine(vettoriale)	convertitore di file PostScript e PDF in grafica vettoriale modificabile (SVG)
libwmf-bin	V:8, I:191	113	Windows/immagine(vettoriale)	strumenti di conversione di metafile windows (dati di grafica vettoriale)
fig2sxd	V:0, I:0	149	fig → sxd(vettoriale)	converte file XFig nel formato di Draw di OpenOffice.org
unpaper	V:2, I:19	460	immagine → immagine	strumento di post-elaborazione per pagine scansionate per OCR
tesseract-ocr	V:8, I:37	1507	immagine → testo	software OCR libero basato sul motore OCR commerciale di HP
tesseract-ocr-eng	V:6, I:37	4032	immagine → testo	dati per motore OCR: file di lingua tesseract-ocr per testi in inglese
gocr	V:1, I:11	531	immagine → testo	software OCR libero
ocrad	V:0, I:4	303	immagine → testo	software OCR libero
eog	V:63, I:253	10189	immagine(Exif)	visualizzatore di immagini grafiche Eye of GNOME
gthumb	V:5, I:21	5483	immagine(Exif)	navigatore e visualizzatore di immagini (GNOME)
geeqie	V:6, I:19	14643	immagine(Exif)	visualizzatore d'immagini che usa GTK
shotwell	V:16, I:217	6451	immagine(Exif)	organizzatore di foto digitali (GNOME)
gtkam	V:0, I:5	1154	immagine(Exif)	applicazione per recuperare dati multimediali da fotocamere digitali (GTK)
gphoto2	V:0, I:12	955	immagine(Exif)	client gphoto2 a riga di comando per le fotocamere digitali
gwenview	V:27, I:94	10651	immagine(Exif)	visualizzatore di immagini (KDE)
kamera	I:93	798	immagine(Exif)	supporto per fotocamere digitali in applicazioni KDE
digikam	V:2, I:12	2646	immagine(Exif)	applicazione per la gestione delle foto digitali per KDE
exiv2	V:2, I:40	321	immagine(Exif)	strumento di manipolazione di metadati EXIF/IPTC
exiftran	V:1, I:19	70	immagine(Exif)	trasforma immagini JPEG di fotocamere digitali
jhead	V:1, I:10	110	immagine(Exif)	manipola la parte non-immagine di file JPEG (fotografie digitali) aderenti conformi ad Exif
exif	V:1, I:12	339	immagine(Exif)	utilità a riga di comando per mostrare informazioni EXIF in file JPEG
exiftags	V:0, I:4	292	immagine(Exif)	utilità per leggere i tag EXIF da un file JPEG di una fotocamera digitale
exifprobe	V:0, I:4	499	immagine(Exif)	legge metadati da immagini digitali
dcraw	V:2, I:18	535	immagine(Raw)	depedifica immagini raw di fotocamere digitali
findimagedupes	V:0, I:1	79	immagine → immagine	trova immagini simili visivamente o duplicati
ale	V:0, I:0	839	immagine → immagine	fonde immagini per migliorarne la fedeltà o creare mosaici
imageindex	V:0, I:2	145	immagine(Exif)	genera gallerie HTML statiche da immagini
outguess	V:0, I:2	261	jpeg.png	strumento steganografico universale
librecad	V:3, I:18	7893	DXF	editor di dati CAD (KDE)
blender	V:3, I:37	79166	blend, TIFF, VRML, ...	editor di contenuti 3D per animazioni, ecc.
blender	V:3, I:37	79166	ms3d, obi,	...

Suggerimento

Qualsiasi fotocamera digitale o scanner con un supporto di archiviazione removibile funziona in Linux attraverso i lettori di [archiviazione USB](#) dato che segue le [regole progettuali per i file system delle fotocamere](#) e usa il file system [FAT](#). Vedere Sezione [10.1.7](#).

11.7 Conversioni di dati vari

Ci sono molti altri programmi per convertire dati. I pacchetti seguenti, trovati usando l'espressione regolare "~Guse::converting" in `aptitude(8)` (vedere Sezione [2.2.6](#)), hanno catturato la mia attenzione.

pacchetto	popcon	dimensione	parola chiave	descrizione
alien	V:2, I:31	163	rpm/tgz → deb	convertitore di pacchetti estranei in pacchetti Debian
freepwing	V:0, I:0	421	EB → EPWING	convertitore da "Electric Book" (popolare in Giappone) in un singolo formato JIS X 4081 (un sottoinsieme di EPWING V1)
calibre	V:8, I:35	55048	tutto → EPUB	convertitore e gestione di una biblioteca per libri elettronici

Tabella 11.18: Elenco di strumenti di conversione di dati vari

Si possono estrarre i dati dal formato RPM anche nel modo seguente.

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

Capitolo 12

Programmazione

Vengono forniti in questo capitolo alcune informazioni base da cui partire per imparare a programmare su un sistema Debian abbastanza da seguire il codice sorgente impacchettato. Quello che segue è un elenco dei pacchetti importanti per la programmazione e dei corrispettivi pacchetti di documentazione.

Guide di riferimento in linea sono disponibili digitando "man nome" dopo aver installato i pacchetti `manpages` e `manpages-dev`. Le guide di riferimento in linea per gli strumenti GNU sono disponibili digitando "info nome_programma", dopo aver installato i pertinenti pacchetti di documentazione. Può essere necessario includere gli archivi "contrib" e "non-free", oltre all'archivio `main`, dato che alcune documentazioni GFDL non sono considerate conformi alle DFSG.

Please consider to use version control system tools. See Sezione [10.5](#).



avvertimento

Non usare "test" come nome di un file di prova eseguibile. "test" è un comando interno della shell.



Attenzione

I programmi software compilati direttamente dai sorgenti andrebbero installati in "/usr/local" o "/opt" per evitare conflitti.

Suggerimento

[Esempi di codice per creare la "Canzone 99 bottiglie di birra"](#) dovrebbe dare buone idee per praticamente tutti i linguaggi di programmazione.

12.1 Script shell

Uno [script di shell](#) è un file di testo con il bit di esecuzione impostato e contiene i comandi nel formato seguente.

```
#!/bin/sh
... command lines
```

La prima riga specifica l'interprete di shell che legge ed esegue il contenuto di questo file.

Leggere script di shell è il modo **migliore** per capire come funziona un sistema *nix. In questa sezione vengono forniti alcune nozioni di riferimento e promemoria per la programmazione di shell. Vedere "Errori in shell" (<http://www.greenend.org.uk/rjk/2001/04/shell.html>) per imparare dagli errori.

A differenza della modalità interattiva della shell (vedere Sezione [1.5](#) e Sezione [1.6](#)), gli script di shell usano spesso parametri, costrutti condizionali e cicli.

12.1.1 Compatibilità con la shell POSIX

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

Evitare di scrivere uno script di shell con **bashismi** o **zshismi** per renderlo portabile tra tutte le shell POSIX. Si può controllare uno script con `checkbashisms(1)`.

Buono: POSIX	Da evitare: bashismo
<code>if ["\$pippo" = "\$pluto"] ; then ...</code>	<code>if ["\$pippo" == "\$pluto"] ; then ...</code>
<code>diff -u file.c.orig file.c</code>	<code>diff -u file.c{.orig,}</code>
<code>mkdir /pippopluto /pippopaperino</code>	<code>mkdir /pippo{pluto,paperino}</code>
<code>nomefunzione() { ...}</code>	<code>function nomefunzione() { ...}</code>
formato ottale: <code>"\377"</code>	formato esadecimale: <code>"\xff"</code>

Tabella 12.1: Elenco di bashismi tipici

Il comando `echo` deve essere usato con le precauzioni seguenti dato che la sua implementazione è diversa negli svariati comandi interni della shell ed esterni.

- Evitare l'uso di qualsiasi opzione di comando tranne `-n`.
- Evitare l'uso di sequenze di escape nelle stringhe dato che la loro gestione è variabile.

Nota

Sebbene l'opzione `-n` **non** faccia veramente parte della sintassi POSIX, è generalmente accettata.

Suggerimento

Se è necessario inserire sequenze di escape nella stringa in output, usare il comando `printf` al posto del comando `echo`.

12.1.2 Parametri di shell

Negli script di shell vengono spesso usati parametri speciali.

Le nozioni base da ricordare riguardanti la **espansione dei parametri** sono le seguenti.

I due punti `:"` in tutti gli operatori nell'elenco precedente sono di fatto opzionali.

- **con** `:"` l'operatore = controlla che il suo operando **esista** e sia **non nullo**
 - **senza** `:"` l'operatore = controlla solo che il suo operando **esista**
-

parametro di shell	valore
\$0	nome della shell o dello script di shell
\$1	primo (1°) argomento di shell
\$9	nono (9°) argomento di shell
\$#	numero di parametri posizionali
"\$*"	"\$1 \$2 \$3 \$4 ..."
"\$@"	"\$1" "\$2" "\$3" "\$4" ...
\$?	stato d'uscita del comando più recente
\$\$	PID dello script di shell
\$!	PID del compito sullo sfondo avviato più recentemente

Tabella 12.2: Elenco di parametri di shell

forma della espressione con parametri	valore se var è impostata	valore se var non è impostata
<code>\${var:-stringa}</code>	"\$var"	"stringa"
<code>\${var:+stringa}</code>	"stringa"	"null"
<code>\${var:=stringa}</code>	"\$var"	"stringa" (ed esegue "var=stringa")
<code>\${var:?string}</code>	"\$var"	invia con echo "stringa" allo stderr (ed esce con stato di errore)

Tabella 12.3: Elenco di espansioni di parametri di shell

forma della sostituzione di parametri	risultato
<code>\${var%suffisso}</code>	rimuove il più piccolo modello di suffisso
<code>\${var%%suffisso}</code>	rimuove il più grande modello di suffisso
<code>\${var#prefisso}</code>	rimuove il più piccolo modello di prefisso
<code>\${var##prefisso}</code>	rimuove il più grande modello di prefisso

Tabella 12.4: Elenco di sostituzioni chiave di parametri di shell

12.1.3 Costrutti condizionali della shell

Ogni comando restituisce uno **stato di uscita** che può essere usato in costrutti condizionali.

- Successo: 0 ("Vero")
- Errore: non 0 ("Falso")

Nota

"0" nel contesto condizionale della shell significa "Vero", mentre "0" nel contesto condizionale in C significa "Falso".

Nota

"[" è l'equivalente del comando `test` che valuta i propri argomenti sino a "]" come un'espressione condizionale.

Le **espressioni condizionali** di base che è bene ricordare sono le seguenti.

- `"comando && se_successo_esegue_anche_questo_comando || true"`
- `"comando || se_non_successo_esegue_anche_questo_comando || true"`
- Una porzione su più righe di script come la seguente

```
if [ conditional_expression ]; then
  if_success_run_this_command
else
  if_not_success_run_this_command
fi
```

In questo caso il `" || true"` finale era necessario per assicurare che lo script non termini accidentalmente a tale riga quando la shell è invocata con l'opzione `"-e"`.

equazione	condizione perché venga restituito il valore logico "vero"
<code>-e file</code>	<i>file</i> esiste
<code>-d file</code>	<i>file</i> esiste ed è una directory
<code>-f file</code>	<i>file</i> esiste ed è un file regolare
<code>-w file</code>	<i>file</i> esiste ed è scrivibile
<code>-x file</code>	<i>file</i> esiste ed è eseguibile
<code>file1 -nt file2</code>	<i>file1</i> è più recente di <i>file2</i> (data di modifica)
<code>file1 -ot file2</code>	<i>file1</i> è più vecchio di <i>file2</i> (data di modifica)
<code>file1 -ef file2</code>	<i>file1</i> e <i>file2</i> sono sullo stesso device e stesso numero inode

Tabella 12.5: Elenco di operatori per paragonare file in espressioni condizionali

Gli operatori **aritmetici** di comparazione di interi nelle espressioni condizionali sono `"-eq"`, `"-ne"`, `"-lt"`, `"-le"`, `"-gt"` e `"-ge"`.

12.1.4 Cicli di shell

Ci sono diverse espressioni per cicli usabili nella shell POSIX.

equazione	condizione perché venga restituito il valore logico "vero"
<code>-z str</code>	la lunghezza di <code>str</code> è zero
<code>-n str</code>	la lunghezza di <code>str</code> è diversa da zero
<code>str1 = str2</code>	<code>str1</code> e <code>str2</code> sono uguali
<code>str1 != str2</code>	<code>str1</code> e <code>str2</code> non sono uguali
<code>str1 < str2</code>	se ordinate, <code>str1</code> viene prima di <code>str2</code> (dipendente dalla localizzazione)
<code>str1 > str2</code>	se ordinate, <code>str1</code> viene dopo di <code>str2</code> (dipendente dalla localizzazione)

Tabella 12.6: Elenco di operatori per paragonare stringhe in espressioni condizionali

- `"for x in pippo1 pippo2 ...; do comando ; done"` ripete il ciclo assegnando gli elementi nell'elenco `"pippo1 pippo2 ..."` alla variabile `"x"` ed eseguendo `"comando"`.
- `"while condizione ; do comando ; done"` ripete `"comando"` fintanto che `"condizione"` è vera.
- `"until condizione ; do comando ; done"` ripete `"comando"` fintanto che `"condition"` è non vera.
- `"break"` permette di uscire dal ciclo.
- `"continue"` permette di riprendere dalla successiva iterazione del ciclo.

Suggerimento

L'iterazione numerica in stile linguaggio C può essere realizzata usando `seq(1)` come `"pippo1 pippo2 ..."`

Suggerimento

Vedere Sezione [9.4.9](#).

12.1.5 La sequenza di elaborazione della riga di comando di shell

A grandi linee la shell elabora uno script nel modo seguente.

- La shell legge una riga.
- La shell raggruppa parte della riga come **un unico elemento** se è racchiusa in `"..."` o `'...'`.
- La shell spezza le altre parti della riga in **elementi** in base ai caratteri seguenti.
 - Spazi bianchi: *spazio tabulazione* <a capo>
 - Metacharacters: `| ; & ()`
- La shell controlla, per ciascun elemento non racchiuso tra `"..."` o `'...'`, la presenza di **parole riservate** per regolare il proprio comportamento.
 - **Parole riservate:** `if then elif else fi for in while unless do done case esac`
- La shell espande gli **alias** se non sono racchiusi in `"..."` o `'...'`.
- La shell espande il carattere **tilde** se non è racchiuso in `"..."` o `'...'`.
 - `"~"` → directory home dell'utente attuale
 - `"~utente"` → directory home di *utente*
- La shell espande **parametri** nei loro valori, se non sono racchiusi in `'...'`.
 - **Parametro:** `"$PARAMETRO"` o `"${PARAMETRO}"`

- La shell espande **sostituzioni di comandi**, se non sono racchiuse in `'...'`.
 - `"$(comando)"` → output di "comando"
 - `"` command `"` → output di "comando"
- La shell espande **glob di nomi percorso** nei nomi di file corrispondenti, se non sono racchiusi in `"..."` o `'...'`.
 - `*` → qualsiasi carattere
 - `?` → un carattere
 - `[...]` → uno qualunque dei caratteri in `"..."`
- La shell cerca **comando** tra le cose seguenti e lo esegue.
 - Definizione di **funzione**
 - comando **interno**
 - **file eseguibile** in `"$PATH"`
- La shell si sposta alla riga seguente e ripete nuovamente questo processo dall'inizio di questa sequenza.

Virgolette singole all'interno di virgolette doppie non hanno alcun effetto.

L'esecuzione di `"set -x"` nella shell o l'invocazione della shell con l'opzione `"-x"` fanno sì che la shell stampi tutti i comandi eseguiti. Ciò è piuttosto utile per il debug.

12.1.6 Programmi di utilità per script di shell

Per far sì che il proprio programma di shell sia il più portabile possibile tra i sistemi Debian, è una buona idea limitare i programmi di utilità a quelli forniti dai pacchetti **essenziali**.

- `"aptitude search ~E"` elenca i pacchetti **essenziali**.
- `"dpkg -L nome_pacchetto |grep '/man/man.*/'"` elenca le pagine man per i comandi forniti dal pacchetto `nome_pacchetto`.

pacchetto	popcon	dimensione	descrizione
dash	V:913, I:993	221	small and fast POSIX-compliant shell for sh
coreutils	V:901, I:999	17478	Utilità GNU di base
grep	V:837, I:999	1091	GNU grep, egrep and fgrep
sed	V:834, I:999	883	GNU sed
mawk	V:392, I:997	242	small and fast awk
debianutils	V:932, I:999	230	utilità varie specifiche di Debian
bsdutils	V:708, I:999	394	utilità di base per 4.4BSD-Lite
bsdextrautils	V:80, I:103	384	extra utilities from 4.4BSD-Lite
moreutils	V:12, I:37	237	utilità Unix aggiuntive

Tabella 12.7: Elenco di pacchetti contenenti piccoli programmi di utilità per script di shell

Suggerimento

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 Sezione 1.6 for examples.

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

Tabella 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 Sezione 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.

pacchetto	popcon	dimensione	descrizione
x11-utils	V:171, I:581	712	xmessage(1): mostra un messaggio o richiesta in una finestra (X)
whiptail	V:116, I:996	71	mostra riquadri di dialogo amichevoli da script di shell (newt)
dialog	V:15, I:118	1222	mostra riquadri di dialogo amichevoli da script di shell (ncurses)
zenity	V:79, I:393	384	display graphical dialog boxes from shell scripts (GTK)
ssft	V:0, I:0	75	Shell Scripts Frontend Tool, strumento per frontend per script di shell (contenitore per zenity, kdialog, e dialog con gettext)
gettext	V:47, I:302	5843	"/usr/bin/gettext.sh": traduce messaggi

Tabella 12.9: List of dialog programs

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.

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 Sezione 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 Pazzie con corti script 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 }' |
```

Ciò equivale ad una qualsiasi delle righe seguenti.

```
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' |
```

L'ultima è una sorta di indovinello; sfrutta le seguenti caratteristiche di Perl.

- Gli spazi bianchi sono opzionali.
- Esiste una conversione automatica da numero a stringa.
- 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

pacchetto	popcon	dimensione	descrizione
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	scarica le specifiche " The Single UNIX Specifications v2 "
susv3	I:0	16	scarica le specifiche " 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

Tabella 12.10: List of compiler related packages

Here, Sezione [12.3.3](#) and Sezione [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.

12.3.1 C

Si può impostare l'ambiente appropriato per compilare programmi scritti nel [linguaggio di programmazione C](#) nel modo seguente.

```
# apt-get install glibc-doc manpages-dev libc6-dev gcc build-essential
```

Il pacchetto `libc6-dev`, cioè la libreria GNU C, fornisce la [libreria standard C](#) che è una raccolta di file header e routine di libreria usati dal linguaggio di programmazione C.

Vedere come documenti di riferimento per C i seguenti.

- "info libc" (documento di riferimento per le funzioni della libreria C)
- gcc(1) e "info gcc"
- ogni_nome_di_funzione_della_libreria_C(3)
- Kernighan & Ritchie, "The C Programming Language", 2nd edition (Prentice Hall)

12.3.2 Semplice programma in C (gcc)

Un semplice esempio "esempio.c" può essere compilato con una libreria "libm" in un eseguibile "eseg_esempio" nel modo seguente.

```
$ 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
```

In questo esempio, l'uso di "-lm" è necessario per fare il link alla libreria "/usr/lib/libm.so" nel pacchetto libc6 per sqrt(3). La libreria reale è in "/lib/" con nome file "libm.so.6", che è un collegamento simbolico a "libm-2.7.so".

Si guardi l'ultimo elemento nel testo di output: ci sono più di 10 caratteri anche se è stato specificato "%10s".

L'uso di funzioni che operano su puntatori di memoria senza controlli sui limiti, come sprintf(3) e strcpy(3) è deprecato per prevenire exploit di tipo buffer overflow che sfruttano gli effetti di superamento dei limiti di grandezza dei dati.

12.3.3 Flex - un Lex migliorato

Flex è un veloce generatore di [analizzatori lessicali](#) compatibile con Lex.

Un tutorial per flex(1) viene fornito da "info flex".

È necessario fornire i propri "main()" e "yywrap()". Altrimenti il proprio programma flex dovrebbe apparire così per compilare senza una libreria. Questo è dovuto al fatto che "yywrap" è una macro e "%option main" abilita implicitamente "%option noyywrap".

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

In alternativa si può compilare con l'opzione per linker " -lfl" alla fine della propria riga di comando cc(1) (come "-ll" per AT&T-Lex). In questo caso non è necessario usare "%option".

12.3.4 Bison - Yacc migliorato

Svariati pacchetti Debian forniscono un generatore di [parser LR lookahead](#) o [parser LALR](#) compatibile con [Yacc](#).

pacchetto	popcon	dimensione	descrizione
bison	V:9, I:101	2858	generatore GNU di parser LALR
byacc	V:0, I:5	160	generatore Berkeley di parser LALR
btyacc	V:0, I:0	243	generatore di parser backtracking basato su byacc

Tabella 12.11: Elenco di generatori di parser LALR compatibili con Yacc

Un tutorial per `bison(1)` viene fornito da `"info bison"`.

È necessario fornire i propri `"main()"` e `"yyerror()"`. `"main()"` chiama `"yyparse()"` che a sua volta chiama `"yylex()"`, solitamente creato con Flex.

```
%%
```

```
%%
```

12.4 Strumenti di analisi statica del codice

[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.

Suggerimento

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.

12.5 Debug

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

12.5.1 Esecuzione base di gdb

Lo [strumento di debug](#) principale in Debian è `gdb(1)` che permette di ispezionare un programma mentre viene eseguito.

Installare `gdb` e i programmi correlati nel modo seguente.

```
# 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`
-

pacchetto	popcon	dimensione	descrizione
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-re lint	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	strumento di controllo statico del codice Python
perl	V:609, I:992	705	interprete con controllore interno statico del codice: 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	strumento per controllare staticamente la presenza di bug in programmi C
flawfinder	V:0, I:0	181	strumento per esaminare codice sorgente C/C++ e per cercare punti deboli per la sicurezza
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
xmlindent	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

Tabella 12.12: Elenco di strumenti per l'analisi statica del codice

pacchetto	popcon	dimensione	documentazione
gdb	V:12, I:113	9799	"info gdb" fornito da gdb-doc
ddd	V:0, I:10	4092	"info ddd" fornito da ddd-doc

Tabella 12.13: List of debug packages

- [“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
```

Suggerimento

Molti comandi `gdb(1)` possono essere abbreviati. L'espansione del tasto di tabulazione funziona come nella shell.

12.5.2 Fare il debug di pacchetti 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 ./
```

Correggere i bug se necessario.

Spostare la versione del pacchetto ad una che non crei conflitti con le versioni ufficiali di Debian, ad esempio una che termini con “+debug1” quando si ricompilano versioni di cui esiste un pacchetto, o una che termini con “~pre1” quando si ricompilano versioni non ancora rilasciate in pacchetti nel modo seguente.

```
$ dch -i
```

Compilare ed installare i pacchetti con i simboli di debug nel modo seguente.

```
$ export DEB_BUILD_OPTIONS="nostrip noopt"
$ debuild
$ cd ..
$ sudo debi package_name*.changes
```

È necessario controllare gli script di compilazione del pacchetto ed assicurarsi di usare “CFLAGS=-g -Wall” per la compilazione di binari.

12.5.3 Ottenere un backtrace

Quando un programma va in crash, è una buona idea inviare un segnalazione di bug riportando le informazioni di backtrace.

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 Sezione 9.4.12)

Suggerimento

Spesso si vede un backtrace in cui una o più delle prime righe sono in `"malloc()"` o `"g_malloc()"`. Quando ciò accade è probabile che il backtrace non sia molto utile. Il metodo più semplice per trovare informazioni utili è di impostare la variabile d'ambiente `"$MALLOCCHECK_"` al valore 2 (`malloc(3)`). Lo si può fare mentre si esegue `gdb` nel modo seguente.

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

12.5.4 Comandi gdb avanzati

comando	descrizione degli scopi del comando
<code>(gdb) thread apply all bt</code>	ottenere un backtrace per tutti i thread di un programma multi-thread
<code>(gdb) bt full</code>	ottenere i parametri nello stack delle chiamate di funzione
<code>(gdb) thread apply all bt full</code>	ottenere un backtrace e parametri: combinazione delle due opzioni precedenti
<code>(gdb) thread apply all bt full 10</code>	ottenere un backtrace e i parametri per le prime dieci chiamate nello stack per eliminare l'output irrilevante
<code>(gdb) set logging on</code>	scrivere un registro dell'output di <code>gdb</code> in un file (il file predefinito è <code>"gdb.txt"</code>)

Tabella 12.14: Elenco di comandi gdb avanzati

12.5.5 Controllare le dipendenze dalle librerie

Per scoprire le dipendenze di un programma da librerie, usare `ldd(1)` nel modo seguente.

```
$ 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)
```

Affinché `ls(1)` funzioni in un ambiente "chroot", le librerie in questione devono essere disponibili nell'ambiente "chroot".

Vedere Sezione [9.4.6](#).

12.5.6 Dynamic call tracing tools

There are several dynamic call tracing tools available in Debian. See Sezione [9.4](#).

12.5.7 Fare il debug di errori X

Se un programma `gnome preview1` ha ricevuto un errore X, si dovrebbe leggere un messaggio del tipo seguente.

```
The program 'preview1' received an X Window System error.
```

Se ciò avviene, si può provare ad eseguire il programma con "`--sync`" ed interrompere alla funzione "`gdk_x_error`" per ottenere un backtrace.

12.5.8 Strumenti per rilevazione di memory leak

In Debian sono disponibili svariati strumenti di rilevazione di memory leak.

pacchetto	popcon	dimensione	descrizione
libc6-dev	V:241, I:615	14358	<code>mtrace(1)</code> : funzionalità di debug malloc in glibc
valgrind	V:5, I:45	80378	strumento di debug e profilazione per la memoria
electric-fence	V:0, I:5	70	strumento di debug malloc(3)
libdmalloc5	V:0, I:3	393	libreria per il debug dell'allocazione di memoria
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	tracciatore di memory leak per programmi C++

Tabella 12.15: Elenco di strumenti per rilevazione di memory leak

12.5.9 Disassemblatore di binari

Si può disassemblare codice binario con `objdump(1)` nel modo seguente.

```
$ objdump -m i386 -b binary -D /usr/lib/grub/x86_64-pc/stage1
```

Nota

Per disassemblare codice in modo interattivo si può usare `gdb(1)`.

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

Tabella 12.16: List of build tool packages

12.6 Build tools

12.6.1 Make

Make è un'utilità per mantenere gruppi di programmi. Quando `make(1)` viene eseguito legge il file di regole, "Makefile" e aggiorna il file target se dipende da file prerequisiti che sono stati modificati dall'ultima volta che esso stesso è stato modificato oppure se il file target non esiste. L'esecuzione di questi aggiornamenti può avvenire in modo concorrente.

La sintassi del file di regole è la seguente.

```
target: [ prerequisites ... ]
[TAB] command1
[TAB] -command2 # ignore errors
[TAB] @command3 # suppress echoing
```

Qui "[TAB]" è il codice di TAB. Ciascuna riga è interpretata dalla shell dopo la sostituzione delle variabili di make. Usare "\" alla fine di una riga per continuare lo script. Usare "\$\$" per inserire "\$" per valori di ambiente per uno script di shell.

Regole implicite per il target ed i prerequisiti possono essere scritte, per esempio, nel modo seguente.

```
%.o: %.c header.h
```

In questo caso il target contiene il carattere "%" (esattamente un carattere). Il "%" fa corrispondenza con qualsiasi sottostringa non vuota nei nomi di file dei target effettivi. Similmente i prerequisiti usano "%" per mostrare come i loro nomi trovino corrispondenza nei nomi dei target effettivi.

variabile automatica	valore
\$@	target
\$<	primo prerequisito
\$?	tutti i prerequisiti più recenti
\$^	tutti i prerequisiti
\$*	"%" nome base con corrispondenza con il modello target

Tabella 12.17: Elenco di variabili automatiche di make

espansione di variabile	descrizione
<code>pippo1 := pluto</code>	espansione valida una volta sola
<code>pippo2 = pluto</code>	espansione ricorsiva
<code>pippo3 += pluto</code>	accoda

Tabella 12.18: Elenco di espansioni delle variabili di make

Eseguire `make -p -f/dev/null` per vedere le regole interne automatiche.

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 Compilare ed installare un programma



avvertimento

Non sovrascrivere mai file di sistema quando si installano programmi compilati in proprio.

Debian non tocca i file in `"/usr/local/"` o `"/opt"`. Perciò se si compila un programma dai sorgenti, installarlo in `"/usr/local/"` in modo che non interferisca con 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 Disinstallare un programma

Se si hanno i sorgenti originali e questi usano `autoconf(1)/automake(1)`, e se ci si ricorda la configurazione usata, eseguire quanto segue per disinstallare un programma.

```
$ ./configure all-of-the-options-you-gave-it
$ sudo make uninstall
```

In alternativa, se si è assolutamente certi che il processo di installazione mette i file solo in `"/usr/local/"` e lì non c'è nulla di importante, si può cancellare tutto ciò che contiene con la riga di comando seguente.

```
# 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

Pagine web dinamiche interattive di base possono essere create nel modo seguente.

- Le interrogazioni vengono presentate all'utente del browser usando moduli [HTML](#).
- La compilazione e il cliccare sulle voci nel modulo invia una delle stringhe [URL](#) seguenti con i parametri codificati dal browser al web server.
 - `"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"`
- "%nn" nell'URL viene sostituito dal carattere con valore esadecimale nn.
- Viene impostata la variabile d'ambiente: `"QUERY_STRING="VAR1=VAL1 VAR2=VAL2 VAR3=VAL3"`.
- Il programma [CGI](#) (uno qualsiasi dei `"program.*"`) sul server web è eseguito con la variabile d'ambiente `"$QUERY_STRING"`.
- Lo `stdout` del programma CGI viene inviato al browser web ed è presentato come pagina web dinamica interattiva.

Per ragioni di sicurezza è bene non creare a mano nuovi metodi per analizzare i parametri CGI. Per loro esistono moduli Perl e Python comprovati. [PHP](#) è fornito con queste funzionalità. Quando è necessaria l'archiviazione dei dati client vengono usati i [cookie HTTP](#). Quando è necessaria l'elaborazione dei dati lato client, viene spesso usato [Javascript](#).

Per maggiori informazioni vedere [CGI \(Common Gateway Interface\)](#), [Apache Software Foundation](#) e [JavaScript](#).

Cercare "CGI tutorial" su Google digitando l'URL codificato <http://www.google.com/search?hl=en&ie=UTF-8&q=CGI+tutorial> direttamente nell'indirizzo del browser è un buon modo per vedere lo script CGI in azione sul server di Google.

12.8 Traduzione di codice sorgente

Esistono programmi per convertire codice sorgente.

pacchetto	popcon	dimensione	parola chiave	descrizione
perl	V:609, I:992	705	AWK → PERL	converte codice sorgente da AWK a PERL: a2p(1)
f2c	V:0, I:5	442	FORTRAN → C	converte codice sorgente da FORTRAN 77 a C/C++: f2c(1)
intel2gas	V:0, I:0	178	intel → gas	convertitore da NASM (formato Intel) a GAS (GNU Assembler)

Tabella 12.19: Elenco di strumenti per la traduzione di codice sorgente

12.9 Creare pacchetti Debian

Se si desidera creare un pacchetto Debian, leggere i documenti seguenti.

- Capitolo [2](#) per capire le basi del sistema dei pacchetti
- Sezione [2.7.13](#) per capire le basi del processo di port
- Sezione [9.11.4](#) per capire le basi delle tecniche chroot

- `debuild(1)`, and `sbuid(1)`
- Sezione [12.5.2](#) per la ricompilazione per il debug
- [Guide for Debian Maintainers](#) (il pacchetto `debmake-doc`)
- [Debian Developer's Reference](#) (pacchetto `developers-reference`)
- [Manuale Debian Policy](#) (pacchetto `debian-policy`)

Ci sono pacchetti come `debmake`, `dh-make`, `dh-make-perl`, ecc., che aiutano nella creazione dei pacchetti.

Appendice A

Appendice

Ecco il dietro le quinte di questo documento.

A.1 Il labirinto Debian

Il sistema Linux è una piattaforma informatica molto potente per un computer in rete. Tuttavia imparare come usare tutte le sue funzionalità non è semplice. Impostare la coda di stampa LPR con una stampante non PostScript è stato un buon esempio di problema su cui si può inciampare. (Ora non ci sono più problemi dato che le installazioni più recenti usano il nuovo sistema CUPS.)

Esiste una mappa dettagliata completa chiamata "CODICE SORGENTE". È molto accurata, ma estremamente difficile da capire. Esistono anche documenti di consultazione chiamati HOWTO e mini-HOWTO; sono più facili da comprendere ma tendono a dare troppi dettagli e a perdere di vista il quadro generale. Io ho a volte dei problemi a trovare la giusta sezione in un lungo HOWTO quando ho bisogno di trovare un paio di comandi da eseguire.

Spero che questa "Debian Reference (versione 2.87)" (2021-11-03 08:37:36 UTC) possa fornire un buon punto di partenza per le persone nel labirinto Debian.

A.2 Storia del copyright

La Debian Reference è stata iniziata da Osamu Aoki <osamu at debian dot org> come promemoria personale per l'amministrazione di sistema. Molti dei contenuti derivano dalle conoscenze ottenute dalla [mailing-list debian-user](#) e da altre risorse Debian.

Seguendo un suggerimento di Josip Rodin, che è stato molto attivo nel [DDP \(Debian Documentation Project\)](#), è stata creata la "Debian Reference (versione 1, 2001-2007)" come parte della documentazione del DDP.

Dopo 6 anni, Osamu ha realizzato che la "Debian Reference (versione 1)" era sorpassata ed ha iniziato a riscrivere molti dei suoi contenuti. La nuova "Debian Reference (versione 2)" è stata rilasciata nel 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.

Si può rintracciare l'origine e l'ispirazione del tutorial nelle fonti seguenti.

- "[Linux User's Guide](#)" di Larry Greenfield (dicembre 1996)
 - reso obsoleto dal "Debian Tutorial"

- ["Debian Tutorial"](#) di Havoc Pennington. (11 dicembre, 1998)
 - scritto in parte da Oliver Elphick, Ole Tetlie, James Treacy, Craig Sawyer e Ivan E. Moore II
 - reso obsoleto da ["Debian GNU/Linux: Guide to Installation and Usage"](#)
- ["Debian GNU/Linux: Guide to Installation and Usage"](#) di John Goerzen ed Ossama Othman (1999)
 - reso obsoleto dalla ["Debian Reference \(versione 1\)"](#)

Si può rintracciare parte dell'origine e dell'ispirazione per la descrizione dei pacchetti e degli archivi nei documenti seguenti.

- ["Debian FAQ"](#) (versione marzo 2002, quando era mantenuta da Josip Rodin)

Si può rintracciare parte dell'origine e dell'ispirazione per gli altri contenuti nei documenti seguenti.

- ["Debian Reference \(versione 1\)"](#) di Osamu Aoki (2001–2007)
 - resa obsoleta da questa nuova ["Debian Reference \(versione 2\)"](#) nel 2008.

La precedente ["Debian Reference \(versione 1\)"](#) è stata creata con il contributo di molti.

- l'enorme contributo di Thomas Hood sugli argomenti di configurazione della rete
- il significativo contributo di Brian Nelson sugli argomenti relativi ad X ed a VCS
- l'aiuto sugli script build e molte correzioni nei contenuti da parte di Jens Seidel
- l'accurata revisione di David Sewell
- molti aiuti da parte di chi ha tradotto, contribuito e segnalato bug

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.

La descrizione dello strumento di debug gdb è stata estesa usando i [contenuti del Debian wiki sul backtrace](#) con il consenso di Ari Pollak, Loïc Minier e Dafydd Harries.

I contenuti della ["Debian Reference \(versione 2.87\)"](#) (2021-11-03 08:37:36 UTC) sono per la maggior parte miei tranne per quanto specificato in precedenza. Anche questi sono stati aggiornati grazie a contributi.

Il documento ["Debian Reference \(versione 1\)"](#) è stato tradotto da Davide Di Lazzaro (mc0315 at mclink dot it).

Il documento ["Debian Reference \(versione 2\)"](#) è tradotto da Beatrice Torracca (beatricet at libero dot it), usando laddove il testo aveva subito modifiche marginali rispetto alla versione 1 parti del lavoro di traduzione di Davide Di Lazzaro, dopo i necessari cambiamenti. Il documento è stato revisionato da Vittore Luccio (vluccio at gmail dot com).

L'autore, Osamu Aoki, ringrazia tutti coloro che con il loro aiuto hanno reso questo documento possibile.

A.3 Formato del 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.)