



debian

Debian Reference

Copyright © 2013 Osamu Aoki

Questa guida di riferimento Debian, Debian Reference (v2) (2018-01-10 14:50:31 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	10 gennaio 2018	

CRONOLOGIA DELLE REVISIONI

POSIZIONE	DATA	DESCRIZIONE	NOME

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	Il prompt di shell in X	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	7
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	15
1.2.11	File di device speciali	15
1.2.12	procfs e sysfs	16
1.2.13	tmpfs	16
1.3	Midnight Commander (MC)	17

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	18
1.3.5	L'editor interno di MC	18
1.3.6	Il visualizzatore interno di MC	19
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	21
1.4.4	Funzionamento del mouse in stile Unix	21
1.4.5	Il paginatore	21
1.4.6	L'editor di testo	22
1.4.7	Impostare un editor di testi predefinito	22
1.4.8	Personalizzare vim	22
1.4.9	Registrare le attività della shell	23
1.4.10	Comandi Unix di base	23
1.5	Il semplice comando di shell	25
1.5.1	Esecuzione dei comandi e variabili d'ambiente	25
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>	48
2.2.2	Operazioni base per la gestione dei pacchetti dalla riga di comando	49
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>	73
2.7.8	Scaricare ed aggiornare automaticamente i pacchetti	74
2.7.9	Limitare l'uso di banda di APT per gli scaricamenti	74
2.7.10	Retrocessione di emergenza alla versione precedente	74
2.7.11	Chi ha caricato il pacchetto?	75
2.7.12	Il pacchetto equivs	75
2.7.13	Fare il port di un pacchetto nel sistema stabile	76
2.7.14	Server proxy per APT	76
2.7.15	Piccolo archivio pubblico per i pacchetti	77
2.7.16	Registrare e copiare la configurazione di sistema	79
2.7.17	Convertire o installare un pacchetto binario alieno	79
2.7.18	Estrarre pacchetti senza dpkg	79
2.7.19	Ulteriori letture sulla gestione dei pacchetti	80
3	Inizializzazione del sistema	81
3.1	Panoramica del processo di avvio	81
3.1.1	Stadio 1: il BIOS	82
3.1.2	Stadio 2: il bootloader	82
3.1.3	Stadio 3: il mini-sistema Debian	84
3.1.4	Stadio 4: il normale sistema Debian	84
3.2	init in stile SysV	85

3.2.1	Il significato del runlevel	86
3.2.2	La configurazione del runlevel	87
3.2.3	Esempio di gestione dei runlevel	87
3.2.4	Il parametro predefinito per ciascuno script init	88
3.2.5	Il nome host	88
3.2.6	Il filesystem	88
3.2.7	Inizializzazione delle interfacce di rete	88
3.2.8	Inizializzazione dei servizi di rete	89
3.2.9	I messaggi di sistema	89
3.2.10	I messaggi del kernel	89
3.3	Il sistema udev	89
3.3.1	L'inizializzazione dei moduli del kernel	90
4	Autenticazione	92
4.1	Autenticazione Unix normale	92
4.2	Gestire le informazioni su account e password	94
4.3	Password buone	94
4.4	Creare password cifrate	95
4.5	PAM e NSS	95
4.5.1	File di configurazione letti da PAM e NSS	97
4.5.2	La moderna gestione centralizzata del sistema	97
4.5.3	"Perché GNU su non supporta il gruppo wheel"	98
4.5.4	Regole più stringenti per le password	98
4.6	Altri controlli sugli accessi	98
4.6.1	sudo	98
4.6.2	PolicyKit	98
4.6.3	SELinux	99
4.6.4	Limitare l'accesso ad alcuni servizi server	99
4.7	Sicurezza dell'autenticazione	99
4.7.1	Password sicure in Internet	100
4.7.2	Secure Shell, shell sicura	100
4.7.3	Misure aggiuntive di sicurezza per Internet	100
4.7.4	Rendere sicura la password di root	101
5	Impostazione della rete	102
5.1	L'infrastruttura base di rete	102
5.1.1	Risoluzione dei nomi di host	104
5.1.2	Il nome dell'interfaccia di rete	105
5.1.3	L'intervallo degli indirizzi di rete per la LAN	105

5.1.4	Il supporto per i dispositivi di rete	106
5.2	La configurazione moderna della rete per il desktop	106
5.2.1	Strumenti grafici di configurazione della rete	106
5.3	La vecchia connessione e configurazione di rete	107
5.4	Il metodo di connessione alla rete (datato)	108
5.4.1	Connessione DHCP con Ethernet	109
5.4.2	Connessione con IP statico con Ethernet	109
5.4.3	Connessione PPP con pppconfig	109
5.4.4	Connessione PPP alternativa con wvdialconf	110
5.4.5	Connessione PPPoE con pppoeconf	111
5.5	Configurazione base della rete con ifupdown (metodo vecchio)	111
5.5.1	Sintassi di base dei comandi	112
5.5.2	Sintassi di base di <code>"/etc/network/interfaces"</code>	112
5.5.3	Interfaccia di rete loopback	113
5.5.4	Interfaccia di rete servita da DHCP	113
5.5.5	Interfaccia di rete con IP statico	113
5.5.6	Informazioni di base sull'interfaccia LAN wireless	114
5.5.7	Interfaccia LAN wireless con WPA/WPA2	114
5.5.8	Interfaccia LAN wireless con WEP	115
5.5.9	Connessione PPP	115
5.5.10	Connessione PPP alternativa	115
5.5.11	Connessione PPPoE	116
5.5.12	Lo stato di configurazione della rete di ifupdown	116
5.5.13	Riconfigurazione base della rete	116
5.5.14	Pacchetto ifupdown-extra	116
5.6	Configurazione avanzata della rete con ifupdown (strumento datato)	117
5.6.1	Il pacchetto ifplugd	117
5.6.2	Il pacchetto ifmetric	118
5.6.3	L'interfaccia virtuale	118
5.6.4	Sintassi avanzata dei comandi	119
5.6.5	Il blocco di mappatura <code>"mapping"</code>	119
5.6.6	Configurazione della rete commutabile manualmente	120
5.6.7	Uso di script con il sistema ifupdown	122
5.6.8	Mappatura con guessnet	122
5.7	Configurazione della rete a basso livello	123
5.7.1	Comandi iproute2	123
5.7.2	Operazioni sicure a basso livello sulla rete	123
5.8	Ottimizzazione della rete	125
5.8.1	Trovare l'MTU ottimale	125
5.8.2	Impostazione dell'MTU	126
5.8.3	Ottimizzazione di TCP per la WAN	126
5.9	Infrastruttura netfilter	126

6 Applicazioni per la rete	128
6.1 Browser web	128
6.1.1 Configurazione del browser	129
6.2 Il sistema di posta	129
6.2.1 Nozioni di base sulla posta elettronica	130
6.2.2 Nozioni di base sui servizi di posta moderni	130
6.2.3 Strategia di configurazione della posta per postazione PC	131
6.3 Agente di trasporto della posta (MTA)	131
6.3.1 La configurazione di exim4	133
6.3.2 La configurazione di Postfix con SASL	134
6.3.3 La configurazione dell'indirizzo di posta	135
6.3.4 Operazioni di base degli MTA	136
6.4 Programma di posta (MUA, Mail User Agent)	136
6.4.1 Programma di posta base —Mutt	137
6.5 L'utilità per il recupero della posta da remoto e l'inoltro	138
6.5.1 Configurazione di getmail	138
6.5.2 configurazione di fetchmail	139
6.6 Agente di consegna della posta (MDA, Mail Delivery Agent) con filtri	139
6.6.1 configurazione di maildrop	140
6.6.2 configurazione di procmail	141
6.6.3 Riconsegna del contenuto di mbox	142
6.7 Server POP3/IMAP4	142
6.8 Server ed utilità per la stampa	142
6.9 Il server e le utilità per l'accesso remoto (SSH)	143
6.9.1 Nozioni di base su SSH	144
6.9.2 Forwarding della porta per tunnel SMTP/POP3	146
6.9.3 Connettersi senza password remote	146
6.9.4 Gestire client SSH estranei	147
6.9.5 Impostare ssh-agent	147
6.9.6 Spegnerne il sistema remoto su SSH	147
6.9.7 Risolvere il problemi con SSH	147
6.10 Altri server di rete	148
6.11 Altri client di rete	148
6.12 Diagnosi dei demoni di sistema	148

7	Il sistema X Window	150
7.1	Pacchetti fondamentali	150
7.2	Impostare l'ambiente desktop	151
7.2.1	Menu Debian	151
7.2.2	Menu Freedesktop.org	151
7.2.3	Menu Debian dal menu di Freedesktop.org	151
7.3	La relazione client/server	151
7.4	Il server X	151
7.4.1	La (ri)configurazione del server X	152
7.4.2	I metodi di connessione al server X	152
7.5	Avviare il sistema X Window	153
7.5.1	Avviare la sessione X con gdm3	154
7.5.2	Personalizzare la sessione X (metodo classico)	154
7.5.3	Personalizzare la sessione X (metodo nuovo)	154
7.5.4	Connettersi ad un client X remoto via SSH	155
7.5.5	Terminale X sicuro via Internet	155
7.6	Tipi di carattere in X Window	155
7.6.1	Tipi di carattere base	156
7.6.2	Tipi di carattere aggiuntivi	158
7.6.3	Tipi di carattere CJK	158
7.7	Applicazioni per X	159
7.7.1	Applicazioni da ufficio per X	159
7.7.2	Applicazioni di utilità per X	159
7.8	Informazioni varie su X	160
7.8.1	Appunti	160
7.8.2	Mappatura della tastiera e dei pulsanti del dispositivo di puntamento in X	160
7.8.3	Client X classici	161
7.8.4	L'emulatore di terminale per X - xterm	161
7.8.5	Eseguire client X come root	161
8	I18N e L10N	163
8.1	L'input da tastiera	163
8.1.1	Il supporto per metodo di input con IBus	164
8.1.2	Un esempio per il giapponese	164
8.1.3	Disabilitare il metodo di input	165
8.2	L'output sul display	165
8.3	Caratteri dell'Asia dell'est con larghezza ambigua	165
8.4	La localizzazione	165
8.4.1	Nozioni base sulla codifica	166

8.4.2	Logica alla base dell'uso della localizzazione UTF-8	166
8.4.3	La (ri)configurazione della localizzazione	167
8.4.4	Il valore della variabile di ambiente "\$LANG"	167
8.4.5	Localizzazione specifica solamente in X Window	168
8.4.6	Codifica per i nomi di file	168
8.4.7	Messaggi localizzati e documentazione tradotta	169
8.4.8	Effetti della localizzazione	169
9	Suggerimenti per il sistema	170
9.1	Il programma screen	170
9.1.1	Scenario di uso per screen(1)	170
9.1.2	Associazioni dei tasti per il comando screen	171
9.2	Registrazione e presentazione di dati	171
9.2.1	Demoni di registro	171
9.2.2	Analizzatori di registro	171
9.2.3	Registrazione delle attività della shell in modo pulito	172
9.2.4	Visualizzazione personalizzata di dati di testo	173
9.2.5	Visualizzazione personalizzata di date e orari	173
9.2.6	Output colorato per la shell	173
9.2.7	Comandi colorati	174
9.2.8	Registrazione delle attività dell'editor per ripetizioni complesse	174
9.2.9	Registrazione dell'immagine grafica di un'applicazione X	175
9.2.10	Registrazione dei cambiamenti a file di configurazione	175
9.3	Monitorare, controllare ed avviare l'attività dei programmi	175
9.3.1	Cronometrare un processo	175
9.3.2	Priorità di schedulazione	177
9.3.3	Il comando ps	177
9.3.4	Il comando top	177
9.3.5	Elencare i file aperti da un processo	177
9.3.6	Tenere traccia delle attività di un programma	177
9.3.7	Identificazione di processi in base a file o socket	178
9.3.8	Ripetere un comando ad intervalli costanti	178
9.3.9	Ripetere un comando su diversi file	178
9.3.10	Avviare un programma dalla GUI	179
9.3.11	Personalizzare i programmi da avviare	180
9.3.12	Uccidere un processo	181
9.3.13	Pianificare compiti da eseguire una volta sola	181
9.3.14	Pianificare compiti in modo regolare	181
9.3.15	Tasto Alt-SysRq	182

9.4	Suggerimenti per l'amministrazione del sistema	182
9.4.1	Chi è nel sistema?	182
9.4.2	Avvertire tutti gli utenti	183
9.4.3	Identificazione dell'hardware	183
9.4.4	Configurazione dell'hardware	183
9.4.5	Orario di sistema e hardware	183
9.4.6	Configurazione del terminale	184
9.4.7	L'infrastruttura audio	185
9.4.8	Disabilitare il salvaschermo	185
9.4.9	Disabilitare i bip sonori	186
9.4.10	Uso della memoria	186
9.4.11	Verifica della sicurezza e dell'integrità del sistema	187
9.5	Suggerimenti per l'archiviazione dei dati	187
9.5.1	Uso dello spazio su disco	187
9.5.2	Configurazione del partizionamento dei dischi	188
9.5.3	Accedere alle partizioni usando UUID	189
9.5.4	LVM2	189
9.5.5	Configurazione del file system	189
9.5.6	Creare file system e verificarne l'integrità	190
9.5.7	Ottimizzare il file system con opzioni di mount	191
9.5.8	Ottimizzare il file system tramite il superblocco	191
9.5.9	Ottimizzare il disco rigido	192
9.5.10	Ottimizzare le unità a stato solido	192
9.5.11	Usare SMART per prevedere danni ai dischi fissi	193
9.5.12	Specificare una directory per l'archiviazione di dati temporanei usando \$TMPDIR	193
9.5.13	Espandere lo spazio di archiviazione utilizzabile con LVM	193
9.5.14	Espandere lo spazio di archiviazione utilizzabile montando un'altra partizione	194
9.5.15	Espandere lo spazio di archiviazione utilizzabile montando un collegamento ad un'altra directory	194
9.5.16	Espandere lo spazio di archiviazione utilizzabile usando collegamenti simbolici	194
9.5.17	Espandere lo spazio di archiviazione utilizzabile usando overlayfs	195
9.6	Immagine del disco	195
9.6.1	Creare un file con un'immagine di disco	195
9.6.2	Scrivere direttamente sul disco	195
9.6.3	Montare un file con un'immagine di disco	196
9.6.4	Pulire un file con un'immagine di disco	197
9.6.5	Creare un file con immagine di disco vuoto	197
9.6.6	Creare un file con un'immagine ISO9660	198
9.6.7	Scrivere direttamente sul CD/DVD-R/RW	198
9.6.8	Montare un file con un'immagine ISO9660	199

9.7	I dati binari	199
9.7.1	Visualizzare e modificare dati binari	199
9.7.2	Manipolare file senza montare i dischi	199
9.7.3	Dati ridondanti	200
9.7.4	recupero di file dati ed analisi forensi	200
9.7.5	Suddividere un file grande in file più piccoli	200
9.7.6	Pulire il contenuto di file	201
9.7.7	File fittizi	201
9.7.8	Cancellare un intero disco fisso	201
9.7.9	Cancellare area inutilizzate di un disco fisso	202
9.7.10	De-cancellare file cancellati ma ancora aperti	202
9.7.11	Cercare tutti i collegamenti fisici	203
9.7.12	Consumo invisibile dello spazio su disco	203
9.8	Suggerimenti per la cifratura dei dati	203
9.8.1	Cifratura di dischi removibili con dm-crypt/LUKS	204
9.8.2	Cifratura della partizione di swap con dm-crypt	204
9.8.3	Montare dischi removibili con dm-crypt/LUKS	205
9.8.4	Cifratura automatica di file con eCryptfs	205
9.8.5	Montare automaticamente eCryptfs	205
9.9	Il kernel	206
9.9.1	Kernel Linux 2.6/3.x	206
9.9.2	Parametri del kernel	206
9.9.3	Header del kernel	207
9.9.4	Compilare il kernel ed i moduli relativi	207
9.9.5	Compilare i sorgenti del kernel: il metodo raccomandato dal Team del Kernel di Debian	208
9.9.6	Driver per hardware e firmware	208
9.10	Sistema virtualizzato	209
9.10.1	Strumenti per la virtualizzazione	209
9.10.2	Fasi del processo di virtualizzazione	209
9.10.3	Montare il file immagine di disco virtuale	211
9.10.4	Sistema chroot	211
9.10.5	Sistemi desktop multipli	213
10	Gestione dei dati	214
10.1	Condividere, copiare ed archiviare	214
10.1.1	Strumenti di archiviazione e compressione	215
10.1.2	Strumenti di copia e sincronizzazione	215
10.1.3	Esempi di invocazione per archivi	217
10.1.4	Esempi di invocazione per la copia	217

10.1.5	Esempi di invocazione per la selezione di file	218
10.1.6	Supporti di archiviazione	219
10.1.7	Supporti di archiviazione removibili	220
10.1.8	Scelta del file system per la condivisione di dati	221
10.1.9	Condividere dati attraverso una rete	222
10.2	Backup e ripristino	223
10.2.1	Suite con utilità di backup	224
10.2.2	Uno script di esempio per il backup di sistema	226
10.2.3	Uno script di copia per backup di dati	227
10.3	Infrastruttura di sicurezza dei dati	228
10.3.1	Gestione delle chiavi per GnuPG	228
10.3.2	Usare GnuPG su file	229
10.3.3	Usare GnuPG con Mutt	229
10.3.4	Usare GnuPG con Vim	229
10.3.5	Somme di controllo MD5	229
10.4	Strumenti per la fusione di codice sorgente	231
10.4.1	Estrarre differenze da file sorgenti	231
10.4.2	Fondere aggiornamenti per file sorgenti	231
10.4.3	Aggiornare con fusioni a 3 vie	231
10.5	Sistemi di controllo delle versioni	231
10.5.1	Comparazione di comandi VCS	233
10.6	Git	234
10.6.1	Configurazione del client Git	234
10.6.2	Documenti di consultazione per Git	235
10.6.3	Comandi Git	235
10.6.4	Git per repository Subversion	236
10.6.5	Git per registrare la cronologia della configurazione	236
10.7	CVS	237
10.7.1	Configurazione di archivi CVS	237
10.7.2	Accesso locale al CVS	238
10.7.3	Accesso remoto a CVS con pserver	238
10.7.4	Accesso remoto a CVS con ssh	238
10.7.5	Importare un nuovo sorgente in CVS	238
10.7.6	Permessi dei file negli archivi CVS	239
10.7.7	Sequenza di azioni tipiche nell'uso di CVS	239
10.7.8	File più recenti dal CVS	241
10.7.9	Amministrazione del CVS	241
10.7.10	Bit di esecuzione per il checkout CVS	242
10.8	Subversion	242

10.8.1	Configurazione di archivi Subversion	242
10.8.2	Accedere a Subversion tramite il server Apache2	242
10.8.3	Accesso locale a Subversion da parte di un gruppo	243
10.8.4	Accesso remoto a Subversion con SSH	243
10.8.5	Struttura delle directory di Subversion	243
10.8.6	Importare un nuovo sorgente in Subversion	243
10.8.7	Sequenza di azioni tipiche nell'uso di Subversion	244
11	Conversione di dati	247
11.1	Strumenti di conversione di dati testuali	247
11.1.1	Convertire un file di testo con iconv	247
11.1.2	Controllare se un file è in UTF-8 con iconv	249
11.1.3	Convertire nomi di file con iconv	249
11.1.4	Conversione del carattere di fine riga	249
11.1.5	Conversione di tabulazioni	250
11.1.6	Editor con auto-conversione	250
11.1.7	Estrazione del testo puro	251
11.1.8	Evidenziare e formattare dati in puro testo	252
11.2	Dati XML	252
11.2.1	Suggerimenti base per XML	253
11.2.2	Elaborazione XML	254
11.2.3	Estrazione di dati XML	254
11.3	Impaginazione	255
11.3.1	Impaginazione roff	255
11.3.2	TeX/LaTeX	256
11.3.3	Fare una bella stampa di una pagina di manuale	257
11.3.4	Creare una pagina di manuale	257
11.4	Dati stampabili	257
11.4.1	Ghostscript	257
11.4.2	Unire due file PS o PDF	258
11.4.3	Utilità per dati stampabili	258
11.4.4	Stampare con CUPS	259
11.5	Conversione dei dati di posta	259
11.5.1	Nozioni di base sui dati di posta	259
11.6	Strumenti per dati grafici	260
11.7	Conversioni di dati vari	260

12 Programmazione	263
12.1 Script shell	264
12.1.1 Compatibilità con la shell POSIX	264
12.1.2 Parametri di shell	265
12.1.3 Costrutti condizionali della shell	266
12.1.4 Cicli di shell	267
12.1.5 La sequenza di elaborazione della riga di comando di shell	267
12.1.6 Programmi di utilità per script di shell	268
12.1.7 Dialogo di script di shell	269
12.1.8 Esempio di script di shell con zenity	269
12.2 Make	270
12.3 C	271
12.3.1 Semplice programma in C (gcc)	271
12.4 Debug	271
12.4.1 Esecuzione base di gdb	272
12.4.2 Fare il debug di pacchetti Debian	272
12.4.3 Ottenere un backtrace	273
12.4.4 Comandi gdb avanzati	273
12.4.5 Fare il debug di errori X	274
12.4.6 Controllare le dipendenze dalle librerie	274
12.4.7 Strumenti per rilevazione di memory leak	274
12.4.8 Strumenti di analisi statica del codice	274
12.4.9 Disassemblatore di binari	274
12.5 Flex - un Lex migliorato	275
12.6 Bison - Yacc migliorato	275
12.7 Autoconf	275
12.7.1 Compilare ed installare un programma	276
12.7.2 Disinstallare un programma	276
12.8 Pazzie con corti script Perl	276
12.9 Web	277
12.10 Traduzione di codice sorgente	277
12.11 Creare pacchetti Debian	278
A Appendice	279
A.1 Il labirinto Debian	279
A.2 Storia del copyright	279
A.3 Formato del documento	280

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	8
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	20
1.14	Elenco di associazioni di tasti per bash	21
1.15	Elenco delle funzioni del mouse in stile Unix	21
1.16	Elenco di comandi Unix di base	24
1.17	Le 3 parti del valore di localizzazione	26
1.18	Elenco di localizzazioni raccomandate	26
1.19	Elenco di valori di "\$HOME"	28
1.20	Modelli di glob della shell	28
1.21	Codici di uscita dei comandi	29
1.22	Idiomi per comandi di shell	30
1.23	Descrittori di file predefiniti	31
1.24	Metacaratteri per BRE e ERE	33
1.25	L'espressione di sostituzione	34
1.26	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 aptitude(8) e apt-get(8) /apt-cache(8)	49
2.7	Opzioni degne di nota per il comando aptitude(8)	50
2.8	Elenco delle associazioni di tasti per aptitude	51
2.9	Elenco delle viste di aptitude	52
2.10	Organizzazione delle viste standard dei pacchetti	52
2.11	Elenco delle regole per espressioni regolari di aptitude	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 dpkg	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	76
3.1	Elenco di bootloader	83
3.2	Significato dei parametri di GRUB	83
3.3	Elenco di utilità di avvio per il sistema Debian	85
3.4	Elenco dei runlevel e descrizione del loro uso	86
3.5	Elenco dei livelli di errore del kernel	89
4.1	I 3 importanti file di configurazione per pam_unix(8)	92
4.2	Il contenuto della seconda voce di "/etc/passwd"	93
4.3	Elenco di comandi per gestire informazioni su account	94
4.4	Elenco di strumenti per generare password	95
4.5	Elenco dei pacchetti degni di nota per i sistemi PAM e NSS	96
4.6	Elenco di file di configurazione letti da PAM e NSS	96
4.7	Elenco di servizi e porte sicuri e non sicuri	100
4.8	Elenco di strumenti per fornire misure aggiuntive di sicurezza	100
5.1	Elenco degli strumenti di configurazione della rete	103
5.2	Elenco di intervalli di indirizzi di rete	105
5.3	Elenco dei metodi di connessione alla rete e dei percorsi di connessione	108
5.4	Elenco di configurazioni della connessione di rete	108
5.5	Elenco di acronimi per le connessioni di rete	109
5.6	Elenco dei file di configurazione per la connessione PPP con pppconfig	110
5.7	Elenco dei file di configurazione per la connessione PPP con wvdialconf	111
5.8	Elenco dei file di configurazione per la connessione PPPoE con pppoeconf	111

5.9	Elenco di comandi base per la configurazione della rete con ifupdown	112
5.10	Elenco dei blocchi in <code>"/etc/network/interfaces"</code>	112
5.11	Elenco di acronimi per WLAN	114
5.12	Elenco della terminologia per i device di rete	119
5.13	Elenco di comandi avanzati per la configurazione della rete con ifupdown	119
5.14	Elenco delle variabili d'ambiente passate dal sistema ifupdown	122
5.15	Tabella di traduzione dai comandi obsoleti <code>net-tools</code> ai nuovi comandi <code>iproute2</code>	123
5.16	Elenco di comandi di rete a basso livello	124
5.17	Elenco degli strumenti di ottimizzazione della rete	124
5.18	Linee guida di base per il valore di MTU ottimale	125
5.19	Elenco di strumenti per firewall	127
6.1	Elenco di browser web	128
6.2	Elenco di pacchetti con plugin per browser	129
6.3	Elenco di pacchetti base relativi ai server di trasporto della posta per postazioni PC	132
6.4	Elenco delle scelte per l'agente di trasporto della posta (MTA) tra i pacchetti dell'archivio Debian	132
6.5	Elenco delle pagine di manuale di Postfix importanti	134
6.6	Elenco dei file di configurazione correlati all'indirizzo di posta	135
6.7	Elenco di operazioni base degli MTA	136
6.8	Elenco di programmi di posta (MUA)	137
6.9	Elenco di utilità per il recupero della posta da remoto e l'inoltro	138
6.10	Elenco di MDA con filtri	140
6.11	Elenco di server POP3/IMAP4	143
6.12	Elenco di server e utilità di stampa	143
6.13	Elenco dei server e delle utilità per l'accesso remoto	144
6.14	Elenco dei protocolli e dei metodi di autenticazione di SSH	144
6.15	Elenco dei file di configurazione per SSH	145
6.16	Elenco di esempi di avvio di client SSH	145
6.17	Elenco di client SSH per altre piattaforme	147
6.18	Elenco di altri server applicativi di rete	148
6.19	Elenco di altri client applicativi di rete	149
6.20	Elenco di RFC popolari	149
7.1	Elenco di (meta)pacchetti fondamentali per X Window	150
7.2	Elenco della terminologia per server/client	152
7.3	Elenco di metodi di connessione al server X	153
7.4	Tabella dei pacchetti che supportano i sistemi dei tipi di carattere in X Window	156
7.5	Tabella dei tipi di carattere PostScript Type 1	157
7.6	Tabella dei tipi di carattere TrueType	157

7.7	Tabella delle parole chiave usate nei nomi dei tipi di carattere CJK per indicarne il tipo	158
7.8	Elenco di applicazioni base da ufficio per X	159
7.9	Elenco di applicazioni base di utilità per X	160
7.10	Elenco di programmi di base per selezione in X	160
8.1	Elenco dei metodi di input supportati da IBus	164
9.1	Elenco di programmi che supportano connessioni di rete interrotte	170
9.2	Elenco di associazioni di tasti per screen	171
9.3	Elenco di analizzatori del registro di sistema	172
9.4	Esempi di visualizzazione di date e orari per il comando "ls -l" per wheezy	173
9.5	Elenco di strumenti di manipolazione di immagini grafiche	175
9.6	Elenco di pacchetti per registrare la cronologia della configurazione in VCS	175
9.7	Elenco di strumenti per monitorare e controllare l'attività dei programmi.	176
9.8	Elenco di valori di nice per la priorità di schedulazione	177
9.9	Elenco degli stili per il comando ps	177
9.10	Elenco dei segnali usati comunemente con il comando kill	181
9.11	Elenco di tasti per il comando SAK	182
9.12	Elenco di strumenti per l'identificazione dell'hardware	183
9.13	Elenco di strumenti di configurazione dell'hardware	184
9.14	Elenco di pacchetti relativi all'audio	185
9.15	Elenco di comandi per disabilitare il salvaschermo	186
9.16	Elenco di dimensioni della memoria riportate	186
9.17	Elenco di strumenti per verificare la sicurezza e l'integrità del sistema	187
9.18	Elenco di pacchetti di gestione delle partizioni dei dischi	188
9.19	Elenco di pacchetti di gestione dei file system	190
9.20	Elenco di pacchetti che visualizzano e modificano dati binari	199
9.21	Elenco di pacchetti per manipolare file senza montare i dischi	199
9.22	Elenco di strumenti per aggiungere dati ridondanti a file	200
9.23	Elenco di pacchetti per recupero di file dati ed analisi forensi.	200
9.24	Elenco di utilità per la cifratura dei dati	203
9.25	Elenco di pacchetti chiave da installare per la ricompilazione del kernel in un sistema Debian	207
9.26	Elenco di strumenti di virtualizzazione	210
10.1	Elenco di strumenti di archiviazione e compressione	216
10.2	Elenco di strumenti di copia e sincronizzazione	216
10.3	Elenco di possibili scelte per il file system di dispositivi di archiviazione removibili con scenari di uso tipici	221
10.4	Elenco dei servizi di rete da scegliere in base allo scenario di uso tipico	223
10.5	Elenco di suite con utilità di backup	225
10.6	Elenco di strumenti per l'infrastruttura di sicurezza dei dati	228

10.7	Elenco di comandi per GNU Privacy Guard per la gestione delle chiavi	228
10.8	Elenco dei significati dei codici di fiducia	229
10.9	Elenco di comandi GNU Privacy Guard per file	230
10.10	Elenco di strumenti per la fusione di codice sorgente	232
10.11	Elenco di strumenti per sistemi di controllo delle versioni	233
10.12	Comparazione di comandi VCS nativi	234
10.13	Elenco di pacchetti e comandi relativi a Git	236
10.14	Opzioni importanti per comandi CVS (da usare come primi argomenti di cvs(1))	241
10.15	Opzioni importanti per comandi Subversion (da usare come primi argomenti di svn(1))	246
11.1	Elenco di strumenti di conversione di dati testuali	247
11.2	Elenco dei valori delle codifiche e loro uso	248
11.3	Elenco di stili per EOL per differenti piattaforme	250
11.4	Elenco di comandi di conversione di TAB dai pacchetti <code>bsdmainutils</code> e <code>coreutils</code>	250
11.5	Elenco di strumenti per estrarre dati in testo puro	251
11.6	Elenco di strumenti per evidenziare dati in testo puro	252
11.7	Elenco di entità predefinite per XML	253
11.8	Elenco di strumenti XML	254
11.9	Elenco di strumenti DSSSL	254
11.10	Elenco di strumenti di estrazione di dati XML	255
11.11	Elenco di strumenti per belle stampe XML	255
11.12	Elenco di strumenti per impaginazione	255
11.13	Elenco di pacchetti che aiutano a creare una pagina man	257
11.14	Elenco di interpreti PostScript Ghostscript	257
11.15	Elenco di utilità per dati stampabili	258
11.16	Elenco di pacchetto che aiutano a convertire dati di posta	259
11.17	Elenco di strumenti per dati grafici	261
11.18	Elenco di strumenti di conversione di dati vari	262
12.1	Elenco di pacchetti di aiuto per la programmazione	263
12.2	Elenco di bashismi tipici	265
12.3	Elenco di parametri di shell	265
12.4	Elenco di espansioni di parametri di shell	265
12.5	Elenco di sostituzioni chiave di parametri di shell	266
12.6	Elenco di operatori per paragonare file in espressioni condizionali	266
12.7	Elenco di operatori per paragonare stringhe in espressioni condizionali	267
12.8	Elenco di pacchetti contenenti piccoli programmi di utilità per script di shell	268
12.9	Elenco di programmi per interfaccia utente	269
12.10	Elenco di variabili automatiche di make	270

12.11 Elenco di espansioni delle variabili di make	270
12.12 Elenco di comandi gdb avanzati	273
12.13 Elenco di strumenti per rilevazione di memory leak	274
12.14 Elenco di strumenti per l'analisi statica del codice	275
12.15 Elenco di generatori di parser LALR compatibili con Yacc	275
12.16 Elenco di strumenti per la traduzione di codice sorgente	277

Sommario

Questo testo è libero; lo si può ridistribuire 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 \(versione 2\)](#) (2018-01-10 14:50:31 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 software di alta qualità pre-compilati.
- Attenzione alla stabilità e alla sicurezza, con facile accesso agli aggiornamenti di sicurezza.
- Attenzione ad aggiornamenti senza problemi alle più recenti versioni dei software con gli 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).

```
# <comando nell'account root>
$ <comando nell'account utente>
```

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.

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/init.d"`. Ci si senta incoraggiati ad esplorare il suo contenuto digitando quanto segue.

```
$ mc "/etc/init.d/"
```

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/sysv-rc/README.runlevels.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/sysv-rc/README.runlevels.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 2018-01-10 14:37:47 UTC e contengono i dati totali inviati da 196132 rapporti su 168051 pacchetti binari e 28 architetture.

Nota

Notare che l'archivio amd64 unstable contiene attualmente solo 57960 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.

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

All'avvio del sistema, se non è stato installato il [Sistema X Window](#) con un display manager come `gdm3`, viene presentata la schermata di login a caratteri. Supponiamo che il proprio nome host sia `pippo`, il prompt di login apparirà come segue.

```
pippo login:
```

Se è stato installato un ambiente [GUI](#) come [GNOME](#) o [KDE](#), allora ci si può spostare ad un prompt di login premendo `Ctrl-Alt-F1` e si può ritornare all'ambiente grafico con `Alt-F7` (vedere Sezione [1.1.6](#) in seguito per maggiori informazioni).

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 jessie/sid pippo tty1
pippo login: pinguino
Password:
Last login: Mon Sep 23 19:36:44 JST 2013 on tty3
Linux snoopy 3.11-1-amd64 #1 SMP Debian 3.11.6-2 (2013-11-01) 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.
pippo:~$
```

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

1.1.2 Il prompt di shell in X

Se è stato installato il [Sistema X Window](#) con un display manager come `gdm3` di [GNOME](#), selezionando l'attività "Ambiente desktop" durante l'installazione, all'avvio del sistema viene presentata una schermata grafica di login. Si inseriscono il nome utente e la password per fare il login come utente non privilegiato. Usare il tasto Tab per spostarsi dal campo del nome utente a quello della password e viceversa, oppure usare il mouse e il clic principale.

Si può ottenere il prompt di shell sotto X avviando un programma **emulatore di terminale** come `gnome-terminal(1)`, `rxvt(1)` o `xterm(1)`. Nell'ambiente desktop GNOME, cliccare su "Applicazioni" → "Accessori" → "Terminale".

Si può anche fare riferimento alla sezione Sezione [1.1.6](#) in seguito.

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.
- Cliccare su "Applicazioni" → "Accessori" → "Terminale Root", nell'ambiente Desktop GNOME.
- 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

Quando il menu del desktop non avvia gli strumenti con interfaccia grafica per l'amministrazione di sistema con i privilegi appropriati, si può avviarli dal prompt di shell di root dell'emulatore di terminale per X, quali `gnome-terminal(1)`, `rxvt(1)` o `xterm(1)`. Vedere Sezione 1.1.4 e Sezione 7.8.5.

**avvertimento**

Non avviare mai il gestore di sessioni/display manager di X dall'account root inserendo `root` al prompt di un display manager come `gdm3(1)`.

**avvertimento**

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.

Se si è nel Sistema X Window, si accedere alla console a caratteri 1 premendo i tasti `Ctrl-Alt-F1`, cioè premendo contemporaneamente il `tasto_Ctrl_sinistro`, il `tasto_Alt_sinistro` e il `tasto_F1`. Si può tornare al Sistema X Window, che di solito è in esecuzione sulla console virtuale 7, premendo `Alt-F7`.

In alternativa ci si può spostare in un'altra console virtuale, per esempio la console 1, dalla riga di comando.

```
# chvt 1
```

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
```

In alternativa, se `/etc/inittab` contiene `ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -h now` al suo interno, per spegnere si può digitare Ctrl-Alt-Canc (il tasto_Ctrl_sinistro, il tasto_Alt_sinistro e Canc sono premuti contemporaneamente). Per i dettagli vedere `inittab(5)`.

Vedere Sezione [6.9.6](#).

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:69, I:240	1440	Gestore di file testuale a tutto schermo
sudo	V:423, I:721	3178	Programma per garantire privilegi di root limitati agli utenti
vim	V:117, I:396	2507	Editor di testi Unix Vi IMproved, un editor di testi per programmatori (versione standard)
vim-tiny	V:65, I:966	1091	Editor di testi Unix Vi IMproved, un editor di testi per programmatori (versione compatta)
emacs25	V:10, I:18	19211	Emacs dal progetto GNU, l'editor di testi estensibile basato su Lisp
w3m	V:184, I:605	2301	Browser Web testuali
gpm	V:13, I:21	489	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.

pacchetto	popcon	dimensione	descrizione
doc-debian	I:856	166	Documentazione del Progetto Debian, (FAQ Debian) ed altri documenti
debian-policy	I:82	5237	Manuale Debian Policy e documenti correlati
developers-reference	I:7	1304	Linee guida ed informazioni per gli sviluppatori Debian
maint-guide	I:4	987	Guida Debian per il nuovo Manutentore
debian-history	I:1	5034	Storia del Progetto Debian
debian-faq	I:845	1277	FAQ Debian

Tabella 1.2: Elenco di pacchetti con documentazione interessante

```
# apt-get install nome_pacchetto
```

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 pesce
```

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 pesce
```

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 "pinguino 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 "pinguino 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**.

**Attenzione**

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)`.

Nota

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
/	la directory root
/etc/	file di configurazione a livello di sistema
/var/log/	file di registro del sistema
/home/	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.
- Nome del **gruppo** a cui il file appartiene.
- **Dimensione** in caratteri (byte) del file.
- **Data ed ora** del file (mtime).
- **Nome** 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"

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 <nuovoproprietario> pippo
# chgrp <nuovogruppo> pippo
# chmod [ugoa][+ -=][rwxXst][, ...] pippo
```

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 /qualche/posizione/
# chown -R pippo:pluto .
# 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 `rw` 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 pippo pluto
$ chmod u=rw,go=r pippo
$ chmod 644 pluto
$ ls -l pippo pluto
-rw-r--r-- 1 pinguino pinguino 0 ott 16 21:39 pluto
-rw-r--r-- 1 pinguino pinguino 0 ott 16 21:35 pippo
```

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)`.

```
(permessi del file) = (permessi del file richiesti) & ~(valore di umask)
```


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**

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.)

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
dialout	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`.

gruppo	comandi accessibili
<code>sudo</code>	eseguire <code>sudo</code> senza password
<code>lpadmin</code>	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

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.

1.2.6 Orari

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

tipo	meaning (historic Unix definition)
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

The actual value of **atime** on GNU/Linux system may be different from that of the historic Unix definition.

- 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.
- Reading a file changes the **atime** attribute of the file on the historic Unix system.
- Reading a file changes the **atime** attribute of the file on the GNU/Linux system if its filesystem is mounted with `"strictatime"`.
- Reading a file for the first time or after one day changes the **atime** attribute of the file on the GNU/Linux system if its filesystem is mounted with `"relatime"`. (default behavior since Linux 2.6.30)
- Reading a file doesn't change the **atime** attribute of the file on the GNU/Linux system if its filesystem is mounted with `"noatime"`.

Nota

The `"noatime"` and `"relatime"` mount options are introduced to improve the filesystem read performance under the normal use cases. Simple file read operation under the `"strictatime"` option accompanies the time-consuming write operation to update the **atime** attribute. But the **atime** attribute is rarely used except for the `mbox(5)` file. See `mount(8)`.

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

Per gli orari, il comando `ls` produce stringhe diverse con le localizzazioni non in inglese ("`it_IT.UTF-8`") rispetto alla vecchia ("`C`").

```
$ LANG=it_IT.UTF-8 ls -l pippo
-rw-rw-r-- 1 pinguino pinguino 0 ott 16 21:35 pippo
$ LANG=C ls -l pippo
-rw-rw-r-- 1 pinguino pinguino 0 Oct 16 21:35 pippo
```

Suggerimento

Vedere Sezione [9.2.5](#) 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 "Contenuto originale" > pluto
$ ls -li pluto
1449840 -rw-rw-r-- 1 penguin penguin 17 ott 16 21:42 pluto
$ ln pluto paper # collegamento fisico
$ ln -s pluto pippo # collegamento simbolico
$ ls -li pluto paper pippo
1449840 -rw-rw-r-- 2 penguin penguin 17 ott 16 21:42 paper
1450180 lrwxrwxrwx 1 penguin penguin 3 ott 16 21:47 pippo -> pluto
1449840 -rw-rw-r-- 2 penguin penguin 17 ott 16 21:42 pluto
$ rm pluto
$ echo "Nuovo contenuto" > pluto
$ ls -li pluto paper pippo
1449840 -rw-rw-r-- 1 penguin penguin 17 ott 16 21:42 paper
1450180 lrwxrwxrwx 1 penguin penguin 3 ott 16 21:47 pippo -> pluto
1450183 -rw-rw-r-- 1 penguin penguin 12 ott 16 21:48 pluto
$ cat paper
Contenuto originale
$ cat pippo
Nuovo contenuto
```

Il collegamento fisico può essere creato all'interno dello stesso file system e condivide uno stesso numero di inode, come rivela l'opzione "`-i`" di `ls(1)`.

Il collegamento simbolico ha sempre permessi di accesso nominali "`lrwxrwxrwx`", 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 si 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 miapipe
$ echo "hello" >miapipe & # messo sullo sfondo
[1] 8022
$ ls -l miapipe
prw-rw-r-- 1 pinguino pinguino 0 ott 16 21:49 miapipe
$ cat miapipe
hello
[1]+  Done                  echo "hello" >miapipe
$ ls miapipe
miapipe
$ rm miapipe
```

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](#)).

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

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.

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 alcune console, come `gnome-terminal(1)`, le battute sui tasti funzione potrebbero essere rubate dal programma di console. Si possono disabilitare queste funzionalità per `gnome-terminal` in "Modifica" → "Scorciatoie da tastiera".

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 [9.4.6](#).

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.)

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

1.3.4 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-"`.

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 nome_del_file_da_modificare
```

```
$ mcedit nome_del_file_da_modificare
```

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-F<n>` 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 percorso/del/nome_file_da_visualizzare
```

```
$ mcview percorso/del/nome_file_da_visualizzare
```

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 [9.3.11](#)). Questa è una funzionalità molto comoda di MC.

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

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)`.

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`.

pacchetto	popcon	dimensione	Shell POSIX	descrizione
bash	V:848, I:999	5798	Sì	Bash : GNU Bourne Again SHell (standard de facto)
tcsh	V:11, I:39	1306	No	TENEX C Shell : una versione migliorata di Berkeley csh
dash	V:925, I:984	204	Sì	Debian Almquist Shell , buona per script di shell
zsh	V:34, I:69	2183	Sì	Z shell : la shell standard con molti miglioramenti
mksh	V:6, I:13	1339	Sì	Una versione della Korn shell
csh	V:2, I:12	339	No	C Shell OpenBSD , una versione di Berkeley csh
sash	V:0, I:5	969	Sì	Stand-alone shell con comandi interni (non adatta per essere la <code>"/bin/sh"</code> standard)
ksh	V:5, I:22	3276	Sì	la versione reale di AT&T della Korn shell
rc	V:0, I:3	154	No	implementazione della rc shell di AT&T Plan 9
posh	V:0, I:0	186	Sì	Policy-compliant Ordinary SHell (derivata da pdksh)

Tabella 1.13: Elenco di programmi shell

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 quando si esce da MC
. /usr/lib/mc/mc.sh

# impostare il CDPATH ad un buon valore
CDPATH=./usr/share/doc:~/~/Desktop:~
export CDPATH

PATH="${PATH:$PATH:}/usr/sbin:/sbin"
# impostare PATH in modo che includa la directory bin privata dell'utente se esiste
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 [9.2.7](#), 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.

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 bash
Ctrl-R	inizia una ricerca incrementale nello storico dei comandi in bash
Tab	completa l'input di un nome di file nella riga di comando in bash
Ctrl-V Tab	inserisce in input un carattere Tab senza espansione nella riga di comando in bash

Tabella 1.14: Elenco di associazioni di tasti per bash

Suggerimento

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

1.4.4 Funzionamento del mouse in stile Unix

Il funzionamento del mouse in stile Unix è basato sul sistema a 3 pulsanti.

azione	risposta
Clic sinistro e trascinamento	seleziona e copia negli appunti
Clic sinistro	marca l'inizio della selezione
Clic destro	marca la fine della selezione e copia negli appunti
Clic di mezzo	incolla gli appunti alla posizione del cursore

Tabella 1.15: Elenco delle funzioni del mouse in stile Unix

La rotella centrale presente nei mouse moderni viene considerata come il pulsante di mezzo e può essere usata per fare il corrispondente clic. Cliccare contemporaneamente con i pulsanti destro e sinistro funziona da clic di mezzo nelle situazioni in cui si hanno mouse a 2 pulsanti. Per poter usare il mouse in una console a caratteri Linux è necessario che `gpm(8)` sia in esecuzione come demone.

1.4.5 Il paginatore

`less(1)` è il paginatore (navigatore del contenuto dei file) migliorato. Premere **h** per l'aiuto. 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/lessf/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-J`.

Nota

I buoni editor, come Vim ed Emacs, possono gestire correttamente testi UTF-8 e testi in altre codifiche esotiche. È opportuno usare l'ambiente X con la localizzazione UTF-8 e installare i programmi e i tipi di carattere necessari. Gli editor hanno opzioni per impostare la codifica dei file indipendentemente dall'ambiente X. Fare riferimento alla loro documentazione sui testi multibyte.

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 9.3.11. Per coerenza, in un sistema Debian, impostare queste variabili a `"/usr/bin/editor"`. (Storicamente `"$EDITOR"` era `"ed"` e `"$VISUAL"` era `"vi"`.)

1.4.8 Personalizzare vim

Si può personalizzare il comportamento di `vim(1)` tramite `"~/.vimrc"`.

Per esempio, provare a fare quanto segue.

```
" -----  
" Configurazione locale  
"  
set nocompatible  
set nopaste  
set pastetoggle=<f2>  
syn on  
if $USER == "root"
```

```
set nomodeline
set noswapfile
else
set modeline
set swapfile
endif
" riempimento per evitare che la riga sopra sia considerata una modeline
" riempimento
" riempimento
```

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.

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

Per esempio, provare a fare quanto segue.

```
$ script
Script è avviato, il file è typescript
```

Eeguire qualsiasi comando di shell sotto `script`.

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

```
$ vim typescript
```

Vedere Sezione [9.2.3](#).

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.

Nota

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

Nota

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.

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

Nota

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).

```
$ <nomecomando> --help
$ <nomecomando> -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.

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.

- Il display manager, come `gdm3`, reimposta variabili d'ambiente.
 - La shell reimposta variabili d'ambiente nel suo codice di avvio in `"~/.bash_profile"` e `"~/.bashrc"`.
-

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 1.17: Le 3 parti del valore di localizzazione

1.5.2 La variabile "\$LANG"

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

Per i codici delle lingue e dei paesi, vedere le descrizioni pertinenti in "info gettext".

In un sistema Debian moderno il codeset dovrebbe essere sempre impostato a **UTF-8**, a meno che non si voglia specificatamente usare uno usato in passato e avendo una buona ragione e le conoscenze per farlo.

Per dettagli più specifici sulla configurazione della localizzazione vedere Sezione 8.4.

Nota

"LANG=en_US" non è "LANG=C" né "LANG=en_US.UTF-8". È "LANG=en_US.ISO-8859-1" (vedere Sezione 8.4.1).

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 1.18: Elenco di localizzazioni raccomandate

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

```
$ date
Sun Jun  3 10:27:39 JST 2007
$ LANG=fr_FR.UTF-8 date
dimanche 3 juin 2007, 10:27:33 (UTC+0900)
```

In questo caso il programma date(1) viene eseguito con valori differenti della variabile d'ambiente "\$LANG".

- Per il primo comando "\$LANG" è impostata al valore di [localizzazione](#) predefinito di sistema: "en_US.UTF-8".
- 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
dimanche 3 juin 2007, 10:27:33 (UTC+0900)
```

Come si può vedere, l'output del comando è in questo caso influenzato dalla variabile d'ambiente per produrre output in francese. Se si desidera che una variabile d'ambiente venga ereditata da sottoprocessi, ad esempio quando si invocano script di shell, è necessario invece **esportarla** nel modo seguente.

```
$ export LANG
```

Nota

Quando si usa un tipico terminale in console, la variabile d'ambiente "\$LANG" è solitamente impostata per essere **esportata** dall'ambiente desktop. Perciò quello sopra non è veramente un buon esempio per testare l'effetto di export.

Suggerimento

Quando si segnala un bug, è una buona idea, se si usa un ambiente non inglese, eseguire e controllare il comando nella localizzazione "LANG=en_US.UTF-8".

Vedere locale(5) e locale(7) per "\$LANG" e variabili d'ambiente correlate.

Nota

Io raccomando di configurare l'ambiente di sistema usando solo la variabile "\$LANG" e di tenersi alla larga dalle variabili "\$LC_*" a meno che non sia strettamente necessario.

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 1.19: 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
dom apr 18 23:22:36 CEST 2010
$ date -R
Sun, 18 Apr 2010 23:23:08 +0200
```

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](#).

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 1.20: Modelli di glob della shell

Per esempio, provare a fare quanto segue.

```
$ mkdir prova; cd prova; 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 prova
```

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 1.21: 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 builtins(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 1.22: 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 >pippo
$ exec 3<pippo 4>pluto # apre i file
$ cat <&3 >&4           # ridirige lo stdin in 3, lo stdout in 4
$ exec 3<&- 4>&-       # chiude i file
$ cat pluto
Hello
```

I descrittori di file 0-2 sono predefiniti.

device	descrizione	descrittore del file
stdin	standard input	0
stdout	standard output	1
stderr	standard error	2

Tabella 1.23: 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 [9.2.7](#).

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.
- Con l'uso di espressioni regolari di base (**BRE**):
 - `grep(1)` trova corrispondenze di testo con modelli.
 - `ed(1)` è un primitivo editor di riga.
 - `sed(1)` è un editor di flussi.
 - `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**):
 - `egrep(1)` trova corrispondenze di testo con modelli.
 - `awk(1)` fa semplici elaborazioni di testo.
 - `tcl(3tcl)` 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)`.
 - `pcregrep(1)` dal pacchetto `pcregrep` 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/ind`

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

Nota

Il criterio di ordinamento e le espressioni di intervalli dipendono dalla localizzazione. Se si desidera ottenere il comportamento tradizionale di un comando, usare la localizzazione **C** invece di quelle **UTF-8** antepo-
nendo al comando `"LANG=C"` (vedere Sezione [1.5.2](#) e Sezione [8.4](#)).

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

Tabella 1.24: 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 [9.2.7](#).

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 \& in emacs)
\n	ciò che ha avuto corrispondenza con la n-esima espressione regolare tra parentesi ("n" è un numero)

Tabella 1.25: 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/DA_REGEX/A_TESTO/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/DA_REGEX/A_TESTO/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/DA_REGEX/A_TESTO/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/DA_REGEX/A_TEST0/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/DA_REGEX/A_TEST0/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   Agosto  1993
Bruce    Perens     Aprile   1996
Ian      Jackson    Gennaio 1998
Wichert  Akkerman   Gennaio 1999
Ben      Collins     Aprile   2001
Bdale    Garbee      Aprile   2002
Martin   Michlmayr   Marzo    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                # mese di inizio
Agosto
Aprile
Gennaio
Gennaio
Aprile
Aprile
Marzo
$ awk '($1=="Ian") { print }' <DPL        # leader del progetto chiamato Ian
Ian      Murdock   Agosto  1993
Ian      Jackson    Gennaio 1998
$ awk '($2=="Perens") { print $3,$4 }' <DPL # quando ha iniziato Perens
Aprile 1996
```

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

Per esempio, provare a fare quanto segue.

```
$ while read nome cognome mese anno; do
    echo $mese
done <DPL
... stesso output del primo esempio Awk
```

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"`.

```
$ vecchioIFS="$IFS" # salva il vecchio valore
$ IFS=':'
$ while read utente password uid gid resto_della_riga; do
    if [ "$utente" = "bozo" ]; then
        echo "l'ID di $utente e' $uid"
    fi
done < /etc/passwd
l'ID di bozo e' 1000
$ IFS="$vecchioIFS" # ripristina il vecchio valore
```

(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=":," # usa ":" e "," come IFS
$ echo IFS=$IFS, IFS="$IFS" # echo e' un comando interno di Bash
IFS= , IFS=;,
$ date -R # un output di comando qualunque
Sat, 23 Aug 2003 08:30:15 +0200
$ echo $(date -R) # sotto-shell --> input alla shell principale
Sat 23 Aug 2003 08 30 36 +0200
$ unset IFS # reimposta il valore predefinito di IFS
$ 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 [9.3.9](#).

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 <code>:</code> (file <code>passwd</code> , 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 1.26: 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: `stretch`.

[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.

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 57960 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)` per tutte le operazioni interattive dalla riga di comando, incluse le installazioni e le rimozioni di pacchetti nonché gli aggiornamenti `dist-upgrade`. Disponibile a partire da Debian Jessie (Debian 8).
- `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
apt	V:879, I:999	3750	Advanced Packaging Tool (APT), frontend per <code>dpkg</code> che fornisce i metodi « <code>http</code> », « <code>ftp</code> » e « <code>file</code> » per l'accesso agli archivi (inclusi i comandi <code>apt/apt-get/apt-cache</code>)
aptitude	V:133, I:723	4412	gestore di pacchetti interattivo basato su terminale con <code>aptitude</code> (8)
tasksel	V:39, I:972	374	strumento per selezionare attività per l'installazione sul sistema Debian (frontend per APT)
unattended-upgrades	V:234, I:388	258	pacchetto di estensione per APT per abilitare l'installazione automatica degli aggiornamenti di sicurezza
dselect	V:4, I:56	2503	gestore di pacchetti basato su terminale (standard precedente, frontend per APT e altri vecchi metodi di accesso)
dpkg	V:940, I:999	6746	sistema di gestione dei pacchetti per Debian
synaptic	V:70, I:446	7793	gestore grafico dei pacchetti (front-end GNOME per APT)
apt-utils	V:423, I:998	1108	programmi di utilità APT: <code>apt-extracttemplates</code> (1), <code>apt-ftparchive</code> (1) e <code>apt-sortpkgs</code> (1)
apt-listchanges	V:364, I:837	383	strumento di notifica dello storico dei cambiamenti dei pacchetti
apt-listbugs	V:8, I:12	449	elenca i bug critici prima di ogni installazione di APT
apt-file	V:13, I:79	82	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

Nonostante le avvertenze dette prima, si sa che molti lettori di questo documento desiderano usare le suite `testing` o `unstable` di Debian come loro sistema principale per **ambienti Desktop auto-amministrati**. Ciò accade perché funzionano molto bene, sono aggiornate di frequente e offrono le più recenti funzionalità.



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, ad esempio il PC di un genitore.

Non serve far altro che impostare semplicemente la stringa relativa alla distribuzione nel file `"/etc/apt/sources.list"` con il nome della suite voluta: `"testing"` o `"unstable"`, o con il suo nome in codice: `"buster"` o `"sid"`. Questo farà vivere una vita di aggiornamenti senza fine.

L'uso di `testing` o `unstable` è **molto divertente** ma comporta qualche rischio. Anche se la suite `unstable` del sistema Debian sembra molto stabile per la maggior parte del tempo, ci sono stati alcuni problemi con i pacchetti delle suite `testing` e `unstable` e qualcuno di questi non era affatto banale da risolvere. Può essere **piuttosto doloroso** per l'utente. A volte si hanno pacchetti con dipendenze non soddisfatte o funzionalità mancanti per qualche settimana.

Ecco alcune idee per assicurare un recupero facile e veloce da bug nei pacchetti Debian.

- 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.
- Creare un ambiente chroot o simile ed eseguire prima lì il sistema più recente (vedere Sezione [9.10](#))

(Se non si è in grado di intraprendere nessuna di queste azioni precauzionali, probabilmente non si è pronti per le suite `testing` e `unstable`.)

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.

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 = stretch`.

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

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

Viene qui usato il nome in codice `stretch` 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 `stretch`, 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 [stretch-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 (stretch)	rilascio stabile (stretch)
http://deb.debian.org/debian/	testing (buster)	rilascio testing (buster)
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	aggiornamenti per il prossimo rilascio stabile minore (opzionale)
http://security.debian.org/	stable/updates	aggiornamenti di sicurezza per il rilascio stabile (importante)
http://security.debian.org/	testing/updates	aggiornamenti di sicurezza per il rilascio testing (importante)
http://deb.debian.org/debian/	stretch-updates	aggiornamenti compatibili per filtri anti-spam, client di messaggistica istantanea, ecc. per stretch
http://deb.debian.org/debian/	stretch-backports	pacchetti più recenti in versione backport per stretch (opzionale)

Tabella 2.2: Elenco dei siti con l'archivio Debian

Suggerimento

Per i sistemi Debian con le suite `stable` e `testing` è una buona norma includere nel file `/etc/apt/sources.list` le righe con `"http://security.debian.org/"` per abilitare gli aggiornamenti di sicurezza, come nell'esempio precedente.

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	57075	aderenti alle DFSG e senza dipendenze da non-free
contrib	339	aderenti alle DFSG ma con dipendenze da non-free
non-free	546	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?](#)

periodo	suite = stable	suite = testing	suite = unstable
dopo il rilascio stretch	nome in codice = stretch	nome in codice = buster	nome in codice = sid
dopo il rilascio buster	nome in codice = buster	nome in codice = bullseye	nome in codice = sid

Tabella 2.4: Relazione tra suite e nome in codice

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".)

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.

Per potere rimanere conforme a questi severi standard per il software libero richiesti per `main`, Debian [una volta toglieva il marchio Mozilla](#) dai pacchetti software Mozilla come Firefox, Thunderbird e Seamonkey, rimuovendo i loro logo e alcuni elementi artistici e li forniva come Iceweasel, Icedove e Iceape rispettivamente. Questi pacchetti sono stati riportati ai loro nomi originali con il rilascio di Debian Stretch (Debian 9) dopo che tali problemi sono stati risolti.

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 invarianti 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 [9.9.6](#) 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 <nome_pacchetto>
$ 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.
-

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)`.

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

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`.

Nota

Dato che `apt` / `apt-get` e `aptitude`, a partire da lenny, condividono le informazioni sui pacchetti installati automaticamente (vedere Sezione 2.5.5), si può mescolare l'uso di questi strumenti senza particolari problemi (vedere il [Bug nr.594490](#)).

«`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.

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"`.

Suggerimento

Il pacchetto `dselect` è ancora disponibile ed è stato lo strumento a tutto schermo interattivo di gestione dei pacchetti dei rilasci passati.

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	stato	descrizione della vista
Vista dei pacchetti	Buona	vedere Tabella 2.10 (vista standard)
Controlla Raccomandati	Buona	elenca i pacchetti che sono raccomandati da alcuni dei pacchetti installati, ma non sono ancora installati
Elenco unico dei pacchetti	Buona	elenca i pacchetti senza organizzarli in gruppi (per l'uso con espressioni regolari)
Consultazione per «Debtags»	Molto usabile	elenca i pacchetti organizzati in base alle loro voci Debtags
Consultazione per categorie	Deprecata	elenca i pacchetti organizzati in base alla loro categoria (usare al suo posto Consultazione per «Debtags»)

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.

- `"~P<termine>" == "~Dprovides:<termine>"`
- `"~C<termine>" == "~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>~n<regex_nome></code>
corrispondenza con la descrizione	<code>~d<regex_descrizione></code>
corrispondenza con il nome del task	<code>~t<regex_task></code>
corrispondenza con il debtag	<code>~G<regex_debtag></code>
corrispondenza con il manutentore	<code>~m<regex_mantainer></code>
corrispondenza con la sezione del pacchetto	<code>~s<regex_sezione></code>
corrispondenza con la versione del pacchetto	<code>~V<regex_versione></code>
corrispondenza con l'archivio	<code>~A{stretch,buster,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 <tipo>	<code>~B<tipogt;</code>
corrispondenza con pacchetti <modello> con dipendenze di tipo <tipo>	<code>~D[<tipo>:]<modello></code>
corrispondenza con pacchetti <modello> con dipendenze non soddisfatte di tipo <tipo>	<code>~DB[<tipo>:]<modello></code>
corrispondenza con pacchetti verso i quali il pacchetto che corrisponde a <modello> dichiara una dipendenza di tipo <tipo>	<code>~R[<tipo>:]<modello></code>
corrispondenza con pacchetti verso i quali il pacchetto <termine> dichiara una dipendenza non soddisfatta di tipo <tipo>	<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 <modello> 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 [9.2.10](#) 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 `”1”` 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 "1" 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 "1" 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 "1" 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 "1" 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 "1" 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 `""stretch""` nel file `"/etc/apt/sources.list"` di esempio descritto in Sezione 2.1.4 con `"buster"` 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 libglb2.0-0:amd64` per elencare il contenuto del pacchetto `libglb2.0-0` 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` (`vederedpkg(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 [9.9](#)).
- Vedere Sezione [12.11](#) 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
COLUMNS=120 dpkg -l <modello_nome_pacchetto>	elenca lo stato di un pacchetto installato per una segnalazione di bug
dpkg -L <nome_pacchetto>	elenca il contenuto di un pacchetto installato
dpkg -L <nome_pacchetto> egrep '/usr/share/man/man.*/.+'	elenca le pagine man di un pacchetto installato
dpkg -S <modello_nome_file>	elenca i pacchetti installati che uno un file con un nome che corrisponde al modello
apt-file search <modello_nome_file>	elenca i pacchetti nell'archivio che hanno un file con un nome che corrisponde al modello
apt-file list <modello_nome_pacchetto>	elenca il contenuto dei pacchetti nell'archivio che corrispondono al modello
dpkg-reconfigure <nome_pacchetto>	riconfigura il pacchetto con esattamente quel nome
dpkg-reconfigure -p=low <nome_pacchetto>	riconfigura il pacchetto con esattamente quel nome ponendo le domande più dettagliate
configure-debian	riconfigura i pacchetti da un'interfaccia a tutto schermo a menu
dpkg --audit	controlla il sistema alla ricerca di pacchetti parzialmente installati
dpkg --configure -a	configura tutti i pacchetti parzialmente installati
apt-cache policy <nome_pacchetto_binario>	mostra la versione disponibile, la priorità e le informazioni nell'archivio di un pacchetto binario
apt-cache madison <nome pacchetto>	mostra la versione disponibile e le informazioni nell'archivio di un pacchetto
apt-cache showsrc <nome_pacchetto_binario>	mostra informazioni sul pacchetto sorgente di un pacchetto binario
apt-get build-dep <nome_pacchetto>	installa i pacchetti richiesti per compilare un pacchetto
aptitude build-dep <nome_pacchetto>	installa i pacchetti richiesti per compilare un pacchetto
apt-get source <nome_pacchetto>	scarica un sorgente (da un archivio standard)
dget <URL del file dsc>	scarica un pacchetto sorgente (da un altro archivio)
dpkg-source -x <nome_pacchetto>_<versione>-<versione_debian>.dsc e "*.*.deb" e "*.*.deb.gz" e "*.*.diff.gz")	crea un albero dei sorgenti a partire da un insieme di pacchetti sorgenti
debuild binary	compila pacchetti da un albero dei sorgenti locale
make-kpkg immagine_kernel	compila un pacchetto kernel a partire da un albero dei sorgenti del kernel
make-kpkg --initrd immagine_kernel	compila un pacchetto kernel con initramfs abilitato, a partire da un albero dei sorgenti
dpkg -i <nome_pacchetto>_<versione>-<versione_debian>_<arch>.deb	installa un pacchetto locale nel sistema
apt install /percorso/di/<nomefile_pacchetto>_<versione>-<versione_debian>_<arch>.deb	installa un pacchetto locale nel sistema, cercando al contempo di risolvere automaticamente le dipendenze
debi <nome_pacchetto>_<versione>-<versione_debian>_<arch>.dsc	installa pacchetti locali nel sistema
dpkg --get-selections '*'> selection.txt	salva le informazioni di stato sui pacchetti selezionati a livello di dpkg
dpkg --set-selections <selection.txt	imposta le informazioni di stato sui pacchetti selezionati a livello di dpkg
echo <nome_pacchetto> hold dpkg --set-selections	imposta lo stato di selezione di un pacchetto a livello di dpkg a hold (equivalente a "aptitude hold<nome_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 `"dists/<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 ripsettiva 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 `stretch-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>_Sources"`
 - `"/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>_Contents"` (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 <code>deb</code>)	<code><nome-pacchetto>_<epoca>:<versione-originale>-<versione></code>
Il pacchetto binario per l'installatore Debian (alias <code>udeb</code>)	<code><nome-pacchetto>_<epoca>:<versione-originale>-<versione></code>
Il pacchetto sorgente (sorgenti originali)	<code><nome-pacchetto>_<epoca>:<versione-originale>-<versione></code>
Il pacchetto sorgente 1.0 (modifiche Debian)	<code><nome-pacchetto>_<epoca>:<versione-originale>-<versione></code>
Il pacchetto sorgente 3.0 (quilt) (modifiche Debian)	<code><nome-pacchetto>_<epoca>:<versione-originale>-<versione></code>
Il pacchetto sorgente (descrizione)	<code><nome-pacchetto>_<epoca>:<versione-originale>-<versione></code>

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	caratteri utilizzabili (espressione regolare)	presenza
<nome-pacchetto>	[a-z,A-Z,0-9,.,+,-]+	richiesta
<epoca>:	[0-9]+:	opzionale
<versione-originale>	[a-z,A-Z,0-9,.,+,-,:]+	richiesta
<versione.debian>	[a-z,A-Z,0-9,.,+,-,~]+	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`.

file	descrizione del contenuto
/var/lib/dpkg/info/<nome_pacchetto>.debconf	file di configurazione (modificabile dall'utente)
/var/lib/dpkg/info/<nome_pacchetto>.deb	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>.conf	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


```
$ ls -l $(type -p vi)
lrwxrwxrwx 1 root root 20 2007-03-24 19:05 /usr/bin/vi -> /etc/alternatives/vi
$ sudo update-alternatives --display vi
...
$ sudo update-alternatives --config vi
Selezione    Percorso
-----
      1      /usr/bin/vim
*+    2      /usr/bin/nvi

Premere Invio per mantenere il valore predefinito[*] o digitare il numero della selezione: ↵
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 spaccetta 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-pa`

```
$ 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 <http://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 /percorso/di/pippo_<vecchia_versione>_<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 /destinazione -i /percorso/di/pippo_<vecchia_versione>_<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 /percorso/del/vecchio/sistema
```

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

```
# cd /percorso/del/vecchio/sistema/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"
- Pacchetti selezionati da pacchetti di dipendenze (es, `python2.4` selezionato da `python`)
- 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 [stretch-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 [9.10](#)) per eseguire programmi da archivi differenti.

2.7.3 Modificare la versione candidata

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**.

**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**

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/stretch-backports/main Packages"`.

2.7.4 Updates e Backports

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

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/ stretch main contrib non-free
deb http://security.debian.org/ stretch/updates main contrib
deb http://deb.debian.org/debian/ stretch-updates main contrib non-free
deb http://deb.debian.org/debian/ stretch-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 `stretch-updates`.
- Solo i vecchi pacchetti installati manualmente da `stretch-backports` vengono aggiornati a quelli più nuovi da `stretch-backports`.

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

```
$ sudo apt-get install -t stretch-backports <nome-pacchetto>
```

2.7.5 Bloccare i pacchetti da installare perché "Raccomandati"

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

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/updates 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 <nome-pacchetto>
```

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: <pacchetto-1>
Pin: release a=unstable
Pin-Priority: 700
```

```
Package: <pacchetto-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/updates 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

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



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/ buster 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/changelog"` 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: wi  
Priority: optional  
Section: admin  
Description-it: aggira le dipendenze dei pacchetti Debian  
Questo pacchetto fornisce uno strumento per creare pacchetti Debian banali.  
Tipicamente questi pacchetti contengono solo informazioni di dipendenza, ma  
possono anche includere file installati normalmente come fanno gli altri  
pacchetti..  
.  
Uno degli usi è quello di creare un metapacchetto: un pacchetto il cui  
unico scopo è di dichiarare le dipendenze e i conflitti con altri  
pacchetti in modo che questi saranno installati, aggiornati o rimossi  
automaticamente.  
.  
Un altro uso è quello di aggirare il controllo delle dipendenze: facendo  
credere a dpkg che sia installata una specifica versione di un dato  
pacchetto quando in realtà non lo è, si possono aggirare dei bug nelle  
dipendenze di altri pacchetti.  
(Si continui a segnalare però questi tipi di bug.)
```

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 pippo
$ apt-get source pippo
$ cd pippo*
```

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 pippo*.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 6.10) come descritto in `apt.conf(5)` ed in `"/usr/share/doc/apt/examples/c".` 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:1	4729	server proxy con cache per i file di archivio Debian (programma OCaml compilato)
apt-cacher	V:1, I:1	289	proxy con cache per file di pacchetto e sorgenti Debian (programma Perl)
apt-cacher-ng	V:4, I:5	1401	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 Piccolo archivio pubblico per i pacchetti

Suggerimento

Impostare manualmente un archivio di pacchetti è complicato. Sono disponibili diversi strumenti di gestione di repository. Un [elenco esaustivo](#) è disponibile online.

Ecco una dimostrazione di come creare manualmente un piccolo archivio pubblico di pacchetti compatibile con il sistema **APT sicuro** (vedere Sezione [2.5.2](#)). Consideriamo la situazione seguente.

- Nome account: "pippo"
- Nome host: "www.esempio.com"
- Pacchetti necessari: apt-utils, gnupg ed altri
- URL: "http://www.esempio.com/~pippo/" (→ "/home/pippo/public_html/index.html")
- Architettura dei pacchetti: "amd64"

Creare una chiave per l'archivio APT di Pippo nel sistema server, nel modo seguente.

```
$ ssh pippo@www.esempio.com
$ gpg --gen-key
...
$ gpg -K
...
sec 1024D/3A3CB5A6 2008-08-14
uid                               Pippo (ARCHIVE KEY) <pippo@www.esempio.com>
ssb 2048g/6856F4A7 2008-08-14
$ gpg --export -a 3A3CB5A6 >pippo.public.key
```

Pubblicare il file della chiave dell'archivio per Pippo, "pippo.public.key", con l'ID di chiave "3A3CB5A6"

Creare un albero di archivio chiamato "Origin: Pippo", nel modo seguente.

```
$ umask 022
$ mkdir -p ~/public_html/debian/pool/main
$ mkdir -p ~/public_html/debian/dists/unstable/main/binary-amd64
$ mkdir -p ~/public_html/debian/dists/unstable/main/source
$ cd ~/public_html/debian
$ cat > dists/unstable/main/binary-amd64/Release << EOF
Archive: unstable
Version: 4.0
Component: main
Origin: Pippo
Label: Pippo
Architecture: amd64
EOF
$ cat > dists/unstable/main/source/Release << EOF
Archive: unstable
Version: 4.0
Component: main
Origin: Pippo
Label: Pippo
Architecture: source
EOF
$ cat >aptftp.conf <<EOF
APT::FTPArchive::Release {
    Origin "Pippo";
    Label "Pippo";
}
```

```

Suite "unstable";
Codename "sid";
Architectures "amd64";
Components "main";
Description "Archivio pubblico di Pippo";
};
EOF
$ cat >aptgenerate.conf <<EOF
Dir::ArchiveDir ".";
Dir::CacheDir ".";
TreeDefault::Directory "pool/";
TreeDefault::SrcDirectory "pool/";
Default::Packages::Extensions ".deb";
Default::Packages::Compress ". gzip bzip2";
Default::Sources::Compress "gzip bzip2";
Default::Contents::Compress "gzip bzip2";

BinDirectory "dists/unstable/main/binary-amd64" {
    Packages "dists/unstable/main/binary-amd64/Packages";
    Contents "dists/unstable/Contents-amd64";
    SrcPackages "dists/unstable/main/source/Sources";
};

Tree "dists/unstable" {
    Sections "main";
    Architectures "amd64 source";
};
EOF

```

Si possono automatizzare aggiornamenti regolari del contenuto dell'archivio APT nel proprio sistema server configurando **dupload**.

Mettere tutti i file pacchetto in "`~/pippo/public_html/debian/pool/main/`" eseguendo "`dupload -t pippo changes`" nel client ed avendo il file "`~/dupload.conf`" che contiene quanto segue.

```

$cfg{'pippo'} = {
    fqdn => "www.esempio.com",
    method => "scpb",
    incoming => "/home/pippo/public_html/debian/pool/main",
    # dinstall su ftp-master invia esso stesso email
    dinstall_runs => 1,
};

$cfg{'pippo'}{postupload}{'changes'} = "
echo 'cd public_html/debian ;
apt-ftparchive generate -c=aptftp.conf aptgenerate.conf;
apt-ftparchive release -c=aptftp.conf dists/unstable >dists/unstable/Release ;
rm -f dists/unstable/Release.gpg ;
gpg -u 3A3CB5A6 -bao dists/unstable/Release.gpg dists/unstable/Release' |
ssh pippo@www.esempio.com 2>/dev/null ;
echo 'Archivio dei pacchetti creato!'"

```

Lo script automatico **postupload** avviato da `dupload(1)` crea file di archivio aggiornati per ciascun caricamento.

Si può aggiungere questo piccolo archivio pubblico nelle righe di apt del proprio sistema client con i comandi seguenti.

```

$ sudo bash
# echo "deb http://www.esempio.com/~pippo/debian/ unstable main" \
    >> /etc/apt/sources.list
# apt-key add pippo.public.key

```

Suggerimento

Se l'archivio si trova sul file system locale, si può usare invece `"deb file:///home/pippo/debian/ ..."`.

2.7.16 Registrare e copiare la configurazione di sistema

Si può creare una copia locale dello stato delle selezioni dei pacchetti e di `debconf` nel modo seguente

```
# dpkg --get-selections '*' > selection.dpkg
# debconf-get-selections > selection.debconf
```

In questo esempio `"*"` fa sì che `"selection.dpkg"` includa anche le voci dei pacchetti "eliminati".

Questi 2 file possono essere trasferiti su un altro computer ed i pacchetti installati lì con i comandi seguenti.

```
# dselect update
# debconf-set-selections < miaselezione.debconf
# dpkg --set-selections < miaselezione.dpkg
# apt-get -u dselect-upgrade # o dselect install
```

Se si ha in progetto di gestire molti server in un cluster con praticamente la stessa configurazione, si dovrebbe considerare l'uso di pacchetti specifici come `fai` per gestire l'intero sistema.

2.7.17 Convertire o installare un pacchetto binario alieno

`alien(1)` permette la conversione di pacchetti binari forniti in file in formato Red Hat `rpm`, Stampede `slp`, Slackware `tgz` e Solaris `pkg` in pacchetti Debian `deb`. Se si vuole usare un pacchetto da una distribuzione Linux diversa da quella installata sul proprio sistema, si può usare `alien` per convertire il formato di pacchetto ed installarlo. `alien` supporta anche pacchetti LSB.

**avvertimento**

`alien(1)` non dovrebbe essere usato per rimpiazzare pacchetti di sistema essenziali, come `sysvinit`, `libc66`, `libpam-modules`, ecc. In pratica, `alien(1)` dovrebbe essere usato solamente per pacchetti solo-binari **non-free** che sono conformi a LSB o con link statico. Per i software liberi si dovrebbero usare i loro pacchetti sorgenti per creare veri pacchetti Debian.

2.7.18 Estrarre pacchetti senza dpkg

In qualsiasi ambiente `*nix` i contenuti di un pacchetto `"dpkg* .deb"` possono essere estratti senza usare `dpkg(1)` utilizzando le utilità standard `ar(1)` e `tar(1)`.

```
# ar x /percorso/di/dpkg_<versione>_<arch>.deb
# ls
total 24
-rw-r--r-- 1 bozo bozo 1320 2007-05-07 00:11 control.tar.gz
-rw-r--r-- 1 bozo bozo 12837 2007-05-07 00:11 data.tar.gz
-rw-r--r-- 1 bozo bozo 4 2007-05-07 00:11 debian-binary
# mkdir control
# mkdir data
# tar xvzf control.tar.gz -C control
# tar xvzf data.tar.gz -C data
```

I contenuti degli altri pacchetti `«*.deb»` possono essere estratti come descritto sopra dal comando `dpkg-deb(1)` ottenuto dal pacchetto `«dpkg* .deb»`, oppure usando in modo simile a prima lo standard `ar(1)` e il più nuovo GNU `tar(1)` con il supporto per la decompressione `xz(1)`.

Si può anche navigare il contenuto del pacchetto usando il comando `mc`.

2.7.19 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(5)` 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:
 - [«Debian New Maintainers' Guide»](#).
 - [«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.

Ho cercato di fornire una veloce panoramica dei punti chiave del sistema Debian e della loro configurazione, come punto di riferimento per l'utente, in base alle conoscenze attuali e passate mie e di altri. Dato che il sistema Debian è in costante evoluzione, la situazione potrebbe essere cambiata. Prima di fare qualsiasi modifica al sistema, si dovrebbe far riferimento alla documentazione più recente per ciascun pacchetto.



avvertimento

Questo capitolo sta diventando non aggiornato dato che è basato su Debian 7.0 (wheezy), rilasciata nel 2013.

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.

Nota

Per piattaforme PC non sorpassate, come il sistema SUN o Macintosh, il BIOS nella ROM e la partizione sul disco possono essere alquanto differenti (Sezione [9.5.2](#)). In questi casi, cercare la documentazione piattaforma specifica altrove.

3.1.1 Stadio 1: il BIOS

Il **BIOS** è il primo stadio del processo di avvio che viene avviato dall'accensione. Il **BIOS**, che risiede nella **ROM (read only memory, memoria in sola lettura)** è eseguito da un particolare indirizzo di memoria al quale è inizializzato, dall'accensione, il contatore di programma della CPU.

Il BIOS effettua l'inizializzazione base dell'hardware (**POST (power on self test, auto-test di accensione)**) e passa il controllo del sistema allo stadio successivo fornito dall'utente. Il BIOS è di solito fornito con l'hardware.

La schermata di avvio del BIOS indica solitamente quale tasto o tasti premere per entrare nella schermata di impostazioni del BIOS, per configurarne il comportamento. Tasti comunemente usati sono F1, F2, F10, Esc, Ins e Canc. Se la schermata di avvio del BIOS è nascosta da una bella schermata grafica, si può premere dei tasti, come Esc, per disabilitarla. Quali tasti vadano premuti dipende fortemente dall'hardware.

Dalla schermata di impostazioni del BIOS si può scegliere la posizione hardware e la priorità del codice avviato dal BIOS. Tipicamente vengono caricati in memoria i primi pochi settori del primo dispositivo selezionato trovato (disco fisso, dischetto floppy, CD-ROM, ...) e viene eseguito questo codice iniziale che può essere una qualsiasi tra le cose seguenti.

- Il codice del bootloader
- Il codice del kernel di SO di passaggio come [FreeDOS](#)
- Il codice del kernel del sistema operativo finale, se può essere contenuto in questo piccolo spazio

Tipicamente il sistema viene avviato dalla partizione specificata del disco fisso primario. I primi 2 settori del disco fisso nei PC vecchi contengono il **master boot record (MBR)**. Le informazioni sulle partizioni del disco, inclusa la selezione per l'avvio sono memorizzate alla fine di questo MBR. Il primo codice boot loader eseguito dal BIOS occupa la parte restante di questo MBR.

3.1.2 Stadio 2: il bootloader

Il **bootloader** è il secondo stadio del processo di avvio che è iniziato dal BIOS. Carica l'immagine kernel del sistema e l'immagine **initrd** in memoria e passa il controllo ad essi. L'immagine **initrd** è l'immagine del file system radice ed il suo supporto dipende dal bootloader usato.

Il sistema Debian normalmente usa il kernel Linux come kernel predefinito del sistema. L'immagine **initrd** dell'attuale kernel Linux 2.6/3.x è tecnicamente l'immagine **initramfs** (initial RAM filesystem). Questa è un archivio **cpio** compresso con **gzip** dei file nel filesystem **root**.

**avvertimento**

Quanto sopra non è più vero con il nuovo **initramfs** multi-segmento. Vedere il [bug n. 790100](#).

L'installazione predefinita del sistema Debian posiziona, per la piattaforma PC, il codice di primo stadio del bootloader GRUB nell'[MBR](#). Sono disponibili molti bootloader e opzioni di configurazione.

**avvertimento**

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

Per GRUB Legacy, il file di configurazione del menu è `"/boot/grub/menu.lst"`. Contiene, per esempio, delle voci come quella seguente.

pacchetto	popcon	dimensioni	initrd	bootloader	descrizione
grub-legacy	V:0, I:2	710	Supportato	GRUB Legacy	Intelligente abbastanza da capire partizioni su disco e filesystem come vfat, ext3, ...
grub-pc	V:31, I:852	578	Supportato	GRUB 2	Intelligente abbastanza da capire partizioni su disco e filesystem come vfat, ext4, ... (predefinito)
grub-rescue-pc	V:0, I:1	6269	Supportato	GRUB 2	È l'immagine di ripristino avviabile di GRUB 2 (CD e floppy) (versione PC/BIOS)
lilo	V:0, I:4	694	Supportato	Lilo	Si basa sulla posizione dei settori dei dati sul disco fisso. (Vecchio)
syslinux	V:5, I:60	299	Supportato	Isolinux	Capisce il filesystem ISO 9660. Usato dal CD di avvio.
syslinux	V:5, I:60	299	Supportato	Syslinux	Capisce il filesystem MSDOS (FAT) . Usato dai dischetti floppy di avvio.
loadlin	V:0, I:1	84	Supportato	Loadlin	Il nuovo sistema viene avviato dal sistema FreeDOS/MSDOS.
mbr	V:1, I:11	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

```

title          Debian GNU/Linux
root           (hd0,2)
kernel        /vmlinuz root=/dev/hda3 ro
initrd        /initrd.img

```

Per GRUB 2, il file di configurazione del menu è `"/boot/grub/grub.cfg"`. Viene generato automaticamente da `"/usr/sbin/update-grub"` usando modelli da `"/etc/grub.d/*"` e impostazioni da `"/etc/default/grub"`. Contiene, per esempio, delle voci come quella seguente.

```

menuentry "Debian GNU/Linux" {
    set root=(hd0,3)
    linux /vmlinuz root=/dev/hda3
    initrd /initrd.img
}

```

Negli esempi precedenti i parametri di GRUB hanno i seguenti significati.

parametro di GRUB	significato
<code>root</code>	usa la terza partizione sul primo disco fisso indicandola come <code>"(hd0, 2)"</code> in GRUB legacy o come <code>"(hd0, 3)"</code> in GRUB 2
<code>kernel</code>	usa il kernel che si trova in <code>"/vmlinuz"</code> con i parametri <code>"root=/dev/hda3 ro"</code>
<code>initrd</code>	usa l'immagine initrd/initramfs che si trova in <code>"/initrd.img"</code>

Tabella 3.2: Significato dei parametri di GRUB

Nota

Il valore per il numero di partizione usato dal programma GRUB Legacy è minore di uno rispetto a quello normale usato dal kernel Linux e dagli strumenti di utilità. Il programma GRUB 2 ha risolto questo problema.

Suggerimento

Per identificare un particolare dispositivo a blocchi si può usare il suo [UUID](#) (vedere Sezione [9.5.3](#)) invece del suo nome file come `"/dev/hda3"`, ad esempio `"root=UUID=81b289d5-4341-4003-9602-e254a17ac232 ro"`.

Suggerimento

Se viene usato [GRUB](#), il parametro di avvio del kernel viene impostato in `/boot/grub/grub.cfg`. Nei sistemi Debian non si dovrebbe modificare direttamente il file `/boot/grub/grub.cfg`. Si dovrebbe modificare il valore `GRUB_CMDLINE_LINUX_DEFAULT` in `/etc/default/grub` ed eseguire `update-grub(8)` per aggiornare `/boot/grub/grub.cfg`.

Suggerimento

Si può avviare un bootloader da un altro bootloader usando una tecnica chiamata [caricamento a catena](#).

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](#).

Lo script `"/init"` viene eseguito come primo programma in questo filesystem root in memoria. È un programma script di shell 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 filesystem root come cifrato.

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)`.

**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»`.

Il programma `init` predefinito ha subito cambiamenti:

- Prima di [squeeze](#) Debian usava il semplice `init` in stile [SysV](#).
- Debian [wheezy](#) migliora l'`init` in stile SysV ordinando la sequenza con intestazioni LSB e avviando gli script di avvio in parallelo.
- Debian [jessie](#) è passata come `init` predefinito a [systemd](#) per l'inizializzazione pilotata da eventi e in parallelo.

Suggerimento

Tutti i meccanismi di avvio sono compatibili grazie agli script `"/etc/init.d/rc"`, `"/etc/init.d/rcS"`, `"/usr/sbin/update-rc.d"` e `"/usr/sbin/invoke-rc.d"`.

Suggerimento

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

pacchetto	popcon	dimensione	descrizione
initscripts	V:319, I:718	205	script per inizializzare ed arrestare il sistema
sysvinit-core	V:14, I:18	225	utilità <code>init(8)</code> in stile System-V
sysv-rc	V:529, I:724	123	meccanismo di cambiamento del runlevel in stile System-V
sysvinit-utils	V:823, I:999	110	utilità in stile System-V (<code>startpar(8)</code> , <code>bootlogd(8)</code> , ...)
lsb-base	V:885, I:999	49	funzionalità di script <code>init</code> Linux Standard Base 3.2
insserv	V:597, I:709	140	strumento per organizzare la sequenza di avvio usando dipendenze LSB negli script <code>init.d</code>
systemd	V:695, I:787	11526	demone <code>init(8)</code> basato su eventi per concorrenza (alternativa a <code>sysvinit</code>)
uswsusp	V:5, I:12	699	strumenti per usare la sospensione software in spazio utente fornita da Linux
kexec-tools	V:1, I:8	269	strumento <code>kexec</code> per riavvii <code>kexec(8)</code> (riavvio a caldo)
bootchart2	V:0, I:1	94	analizzatore delle prestazioni del processo di avvio
pybootchartgui	V:0, I:1	177	analizzatore delle prestazioni del processo di avvio (visualizzazione)
mingetty	V:0, I:3	35	<code>getty(8)</code> solo console
mgetty	V:0, I:1	301	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 in stile SysV



Attenzione

Il sistema Debian predefinito attuale non usa un `init` in stile SysV. Leggere altra documentazione per gli `init` moderni basati su `systemd`. Vedere [The Debian Administrator's Handbook](#).

Questa sezione descrive come il buon vecchio init in stile SysV avviava il sistema. Il sistema Debian non funziona esattamente come descritto qui, ma è piuttosto istruttivo sapere queste nozioni di base dato che i nuovi sistemi init tendono ad offrire funzionalità equivalenti.

Il processo di avvio in stile SysV essenzialmente passa attraverso le seguenti fasi.

1. Il sistema Debian va nel **runlevel N** (None, nessuno) per inizializzare il sistema seguendo la descrizione `"/etc/inittab"`.
2. Il sistema Debian va nel **runlevel S** per inizializzare il sistema in modalità singolo-utente per completare l'inizializzazione dell'hardware, ecc.
3. Il sistema Debian va in uno dei **runlevel multiutente (da 2 a 5)** specificato per avviare i servizi di sistema.

Il runlevel iniziale usato per la modalità multi-utente è specificato con il parametro di avvio del kernel `"init="` o nella riga `"initdefault"` del file `"/etc/inittab"`. Il sistema Debian, nella configurazione predefinita, si avvia nel **runlevel 2**.

Tutti i file script effettivamente eseguiti dal sistema init sono contenuti nella directory `"/etc/init.d/"`.

Vedere `init(8)`, `inittab(5)` e `"/usr/share/doc/sysv-rc/README.runlevels.gz"` per la spiegazione esatta.

3.2.1 Il significato del runlevel

Ogni **runlevel** usa una directory per la propria configurazione e ha un significato specifico come descritto nella tabella seguente.

runlevel	directory	descrizione dell'uso del runlevel
N	nessuna	livello di avvio del sistema (NONE) (nessuna directory <code>"/etc/rcN.d/"</code>)
0	<code>/etc/rc0.d/</code>	arresta il sistema
S	<code>/etc/rcS.d/</code>	modalità utente-singolo all'avvio (alias "S")
1	<code>/etc/rc1.d/</code>	modalità utente-singolo dalla modalità multi-utente
2	<code>/etc/rc2.d/</code>	modalità multi-utente
3	<code>/etc/rc3.d/</code>	""
4	<code>/etc/rc4.d/</code>	""
5	<code>/etc/rc5.d/</code>	""
6	<code>/etc/rc6.d/</code>	riavvia il sistema
7	<code>/etc/rc7.d/</code>	modalità multi-utente valida ma normalmente non usata
8	<code>/etc/rc8.d/</code>	""
9	<code>/etc/rc9.d/</code>	""

Tabella 3.4: Elenco dei runlevel e descrizione del loro uso

Si può cambiare il runlevel dalla console, per esempio a 4, nel modo seguente.

```
$ sudo telinit 4
```



Attenzione

Il sistema Debian non assegna nessuna differenza significativa ai runlevel compresi tra 2 e 5. L'amministratore di un sistema Debian può cambiare questa situazione. (Cioè Debian non è Red Hat Linux né Solaris di Sun Microsystems né HP-UX di Hewlett Packard né AIX di IBM, né ...)



Attenzione

Il sistema Debian non popola le directory per i runlevel tra 7 e 9 durante l'installazione. Le varianti di Unix tradizionali non usano questi runlevel.

3.2.2 La configurazione del runlevel

Quando i comandi `init(8)` o `telinit(8)` vanno nel runlevel "`<n>`", il sistema fondamentalmente esegue gli script di inizializzazione nel modo seguente.

1. Gli script in `"/etc/rc<n>.d/` il cui nome inizia con una "K" vengono eseguiti in ordine alfabetico con l'unico argomento `"stop"`. (uccisione dei servizi)
2. Gli script in `"/etc/rc<n>.d/` il cui nome inizia con una "S" vengono eseguiti in ordine alfabetico con l'unico argomento `"start"`. (avvio dei servizi)

Per esempio, se in una directory di runlevel sono presenti i collegamenti `"S10sysklogd"` e `"S20exim4"`, `"S10sysklogd"`, che è un collegamento simbolico a `"../init.d/sysklogd"`, viene eseguito prima di `"S20exim4"`, che è un collegamento simbolico a `"../init.d/exim4"`.

Questo semplice sistema di inizializzazione sequenziale è il classico sistema di avvio in stile [System V](#) ed è stato usato fino al rilascio Lenny di Debian.

Il sistema Debian recente è stato ottimizzato per eseguire invece gli script di inizializzazione in modo concorrente.

- Il comando `insserv(8)` usa le informazioni di dipendenza dei pacchetti LSB e calcola le dipendenze tra tutti gli script.
 - Vedere `"/usr/share/doc/insserv/README.Debian"`.
- Le informazioni di dipendenza dei pacchetti LSB definite nell'installazione degli script `init` assicura la disponibilità delle risorse necessarie.
 - Vedere [Linux Standard Base Core Specification 3.1, VII. System Initialization](#).
- [Annuncio dell'Abilitazione dell'avvio parallelo come predefinito](#)



avvertimento

È sconsigliato fare qualsiasi cambiamento ai collegamenti simbolici in `"/etc/rcS.d/"` a meno che non si sia più esperti del manutentore.

3.2.3 Esempio di gestione dei runlevel

Per esempio, si può impostare il sistema dei runlevel in un modo che ricorda [Red Hat Linux](#) nel modo seguente.

- `init` avvia il sistema con `runlevel=3` in modo predefinito.
- `init` non avvia `gdm3(1)` nei `runlevel=(0,1,2,6)`.
- `init` avvia `gdm3(1)` nei `runlevel=(3,4,5)`.

Si può ottenere questo modificando con un editor il file `"/etc/inittab"`, per cambiare il runlevel di avvio, e usando strumenti amichevoli di gestione dei runlevel come `sysv-rc-conf` o `bum`, per modificare i runlevel. Se si vuole invece usare solamente la riga di comando, si può farlo nel modo seguente (dopo l'installazione standard del pacchetto `gdm3` e avendolo selezionato come display manager prescelto).

```
# cd /etc/rc2.d ; mv S21gdm3 K21gdm3
# cd /etc ; perl -i -p -e 's/^id:./id:3:/' inittab
```

Notare che il file `"/etc/X11/default-display-manager"` viene controllato quando vengono avviati i demoni per display manager: `xdm`, `gdm3`, `sddm` e `wdm`.

Nota

Si può sempre avviare X da qualsiasi shell in console con il comando `startx(1)`.

3.2.4 Il parametro predefinito per ciascuno script init

Il parametro predefinito per ciascun script init in `/etc/init.d/` è dato dal file corrispondente in `/etc/default/` che contiene **solamente** assegnazioni di variabili d'ambiente. Il nome della directory è una scelta specifica di Debian. È grosso-modo l'equivalente della directory `/etc/sysconfig` di [Red Hat Linux](#) e altre distribuzioni. Per esempio, si può usare `/etc/default/cron` per controllare il funzionamento di `/etc/init.d/cron`.

Il file `/etc/default/rcS` può essere usato per personalizzare impostazioni standard dell'avvio per `motd(5)`, `sulogin(8)`, ecc.

Se non si ottiene il comportamento voluto cambiando queste variabili, si possono [modificare gli script init](#) stessi. Questi sono file di configurazione modificabili dall'amministratore di sistema.

3.2.5 Il nome host

Il kernel gestisce il **nome host** del sistema. Lo script init nel runlevel S, che è un collegamento simbolico a `/etc/init.d/hostname` imposta il nome host del sistema all'avvio (usando il comando `hostname`) 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.

3.2.6 Il filesystem

Benché il filesystem root sia montato dal kernel quando viene avviato, gli altri filesystem sono montati nel runlevel S dagli script init seguenti.

- `/etc/init.d/mountkernfs.sh` per i filesystem del kernel in `/proc`, `/sys`, ecc.
- `/etc/init.d/mountdevsubfs.sh` per i filesystem virtuali in `/dev`
- `/etc/init.d/mountall.sh` per i filesystem normali, usando `/etc/fstab`
- `/etc/init.d/mountnfs.sh` per i filesystem di rete, usando using `/etc/fstab`

Le opzioni usate per montare gli speciali file system del kernel (`procfs`, `sysfs` e `tmpfs` per `/proc`, `/sys`, `/tmp`, `/run`, ecc.) sono impostate in `/etc/default/rcS`. Vedere `rcS(5)`.

Le opzioni usate per montare i file system normali dei dischi e di rete sono impostate in `/etc/fstab`. Vedere Sezione [9.5.7](#).

Nota

L'effettivo montaggio dei filesystem di rete attende l'avvio delle interfacce di rete.



avvertimento

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

3.2.7 Inizializzazione delle interfacce di rete

Le interfacce di rete sono inizializzate nel runlevel S dagli script init che sono collegamenti simbolici a `/etc/init.d/ifupdown-c1` e `/etc/init.d/ifupdown`. Vedere Capitolo [5](#) per la loro configurazione.

3.2.8 Inizializzazione dei servizi di rete

Molti servizi di rete (vedere Capitolo 6) sono avviati nella modalità multi-utente direttamente, come processi demone, all'avvio da script di init, per esempio, `"/etc/rc2.d/S20exim4"` (per `RUNLEVEL=2`) che è un collegamento simbolico a `"/etc/init.d/exim4"`.

Alcuni servizi di rete possono essere avviati a richiesta, usando il [super-server](#) `inetd` (o un equivalente). `inetd` viene avviato all'avvio da `"/etc/rc2.d/S20inetd"` (per `RUNLEVEL=2`) che è un collegamento simbolico a `"/etc/init.d/inetd"`. Fondamentalmente `inetd` permette ad un demone in esecuzione di invocarne svariati altri, riducendo il carico sul sistema.

Quando al [super-server](#) `inetd` arriva una richiesta per un servizio, vengono identificati il suo protocollo e il suo servizio cercandoli nei database in `"/etc/protocols"` e `"/etc/services"`. `inetd` quindi cerca un servizio Internet normale nel database `"/etc/inetd.conf"` o un servizio basato su [Open Network Computing Remote Procedure Call \(ONC RPC\)/Sun RPC](#) in `"/etc/rpc.conf"`.

A volte, `inetd` non avvia il servizio richiesto direttamente, ma avvia il programma [wrapper TCP](#) `tcpd(8)`, con il nome di servizio richiesto, in `"/etc/inetd.conf"`, come argomento. In questo caso, `tcpd` esegue il programma server appropriato dopo aver fatto la registrazione della richiesta e qualche altro controllo addizionale usando `"/etc/hosts.deny"` e `"/etc/hosts.allow"`.

Per la sicurezza del sistema è bene disabilitare più servizi di rete possibile. Vedere Sezione 4.6.4.

Vedere `inetd(8)`, `inetd.conf(5)`, `protocols(5)`, `services(5)`, `tcpd(8)`, `hosts_access(5)`, `hosts_options(5)`, `rpcinfo(8)`, `portmap(8)` e `"/usr/share/doc/portmap/portmapper.txt.gz"`.

3.2.9 I messaggi di sistema

I messaggi di sistema possono essere personalizzati da `"/etc/default/rsyslog"` e `"/etc/rsyslog.conf"` sia per il file di registro sia per la visualizzazione a schermo. Vedere `rsyslogd(8)` e `rsyslog.conf(5)`. Vedere anche Sezione 9.2.2.

3.2.10 I messaggi del kernel

I messaggi del kernel possono essere personalizzati da `"/etc/default/klogd"` sia per il file di registro sia per la visualizzazione a schermo. Impostare `"KLOGD=' -c 3 '"` in tale file ed eseguire `"/etc/init.d/klogd restart"`. Vedere `klogd(8)`.

Si può cambiare direttamente il livello dei messaggi di errore mostrati con il comando seguente.

```
# 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 3.5: Elenco dei livelli di errore del kernel

3.3 Il sistema udev

Per i kernel Linux 2.6 e successivi, [il sistema udev](#) fornisce un meccanismo di rilevazione e inizializzazione automatica dell'hardware (vedere `udev(7)`). Per ogni dispositivo rilevato dal kernel, il sistema `udev` avvia un processo utente che usa le informazioni del filesystem [sysfs](#) (vedere Sezione 1.2.12), carica, usando `modprobe(8)` (vedere Sezione 3.3.1), i moduli del kernel necessari per il supporto del dispositivo e crea i nodi di device corrispondenti.

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`.

Il nome dei nodi di device può essere configurato dai file di regole di `udev` in `/etc/udev/rules.d/`. Le regole predefinite attuali tendono a creare nomi generati dinamicamente che hanno come risultato nomi di device non fissi, tranne che per dispositivi CD e di rete. Aggiungendo le proprie regole personalizzate, in modo simile a quelle per i dispositivi CD e di rete, si possono creare nomi di device statici anche per altri dispositivi, come chiavette USB. Vedere ["Scrivere regole udev"](#) o ["/usr/share/doc/udev/writing_udev_rules/index.html"](#).

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.

Suggerimento

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 [9.5.3](#).

3.3.1 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 [3.3](#)) 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 [5.9](#) 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 [9.4.3](#).

Suggerimento

Si può configurare l'hardware all'avvio per attivare le funzionalità dell'hardware desiderate. Vedere Sezione [9.4.4](#).

Suggerimento

Si può probabilmente aggiungere il supporto per il proprio speciale dispositivo ricompilando il kernel. Vedere Sezione [9.9](#).

Capitolo 4

Autenticazione

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.

**avvertimento**

Questo capito sta diventando non aggiornato dato che è basato su Debian 7.0 (wheezy), rilasciata nel 2013.

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
/etc/passwd	-rw-r--r--	root	root	informazioni sugli account utente (ripulite)
/etc/shadow	-rw-r-----	root	shadow	informazioni sicure sugli account utente
/etc/group	-rw-r--r--	root	root	informazioni sui gruppi

Tabella 4.1: I 3 importanti file di configurazione per `pam_unix(8)`

"/etc/passwd" contiene righe come le seguenti.

```
...
utente1:x:1000:1000:Utente1 Nome,,,:/home/utente1:/bin/bash
utente2:x:1001:1001:Utente2 Nome,,,:/home/utente2:/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.

```
...
utente1:$1$Xop0FYH9$IfxyQwBe9b8tiyIkt2P4F/:13262:0:99999:7:::
utente2:$1$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.

```
gruppo1:x:20:utente1,utente2
```

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
getent passwd <nome_utente>	sfoglia le informazioni sull’account di “<nome_utente>”
getent shadow <nome_utente>	sfoglia le informazioni shadow sull’account di “<nome_utente>”
getent group <nome_gruppo>	sfoglia le informazioni sul gruppo “<nome_gruppo>”
passwd	gestisce la password per l’account
passwd -e	imposta una password usa e getta per l’attivazione dell’account
chage	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 [alioth](#) Debian, il contenuto dei file locali “/etc/passwd”, “/etc/group” e “/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 insieme.

- 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:46, I:670	342	mkpasswd	frontend ricco di funzionalità per la libreria crypt(3)
openssl	V:794, I:990	1303	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.

- 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.

pacchetto	popcon	dimensione	descrizione
libpam-modules	V:825, I:999	874	Pluggable Authentication Modules, moduli di autenticazione inseribili (servizio base)
libpam-ldap	V:2, I:16	244	Pluggable Authentication Module, modulo di autenticazione inseribile, che permette interfacce LDAP
libpam-cracklib	I:17	113	Pluggable Authentication Module, modulo di autenticazione inseribile, che permette il supporto di cracklib
libpam-doc	I:2	1019	Pluggable Authentication Modules, moduli di autenticazione inseribili (documentazione in html e testo semplice)
libc6	V:939, I:999	11065	Libreria GNU C: librerie condivise che forniscono anche il servizio "Name Service Switch"
glibc-doc	I:13	3447	Libreria GNU C: pagine man
glibc-doc-reference	I:5	12982	Libreria GNU C: manuale di riferimento nei formati info, pdf e html (non-free)
libnss-mdns	I:559	107	Modulo NSS per la risoluzione dei nomi DNS Multicast
libnss-ldap	I:14	255	Modulo NSS per l'uso di LDAP come servizio per i nomi
libnss-ldapd	I:19	146	Modulo NSS per l'uso di LDAP come servizio per i nomi (nuovo fork di libnss-ldap)

Tabella 4.5: Elenco dei pacchetti degni di nota per i sistemi PAM e NSS

file di configurazione	funzione
/etc/pam.d/<nome_programma>	imposta la configurazione PAM per il programma "<nome_programma>"; vedere pam(7) e pam.d(5)
/etc/nsswitch.conf	imposta la configurazione NSS con la voce per ciascun servizio. Vedere nsswitch.conf(5)
/etc/nologin	limita il login utente con il modulo pam_nologin(8)
/etc/securetty	limita la tty per l'accesso root con il modulo pam_securetty(8)
/etc/security/access.conf	imposta il limite di accesso con il modulo pam_access(8)
/etc/security/group.conf	imposta restrizioni basate su gruppi con il modulo pam_group(8)
/etc/security/pam_env.conf	imposta le variabili d'ambiente con il modulo pam_env(8)
/etc/environment	imposta variabili d'ambiente aggiuntive con il modulo pam_env(8) con l'argomento "readenv=1"
/etc/default/locale	imposta la localizzazione con il modulo pam_env(8) con l'argomento "readenv=1 envfile=/etc/default/locale" (Debian)
/etc/security/limits.conf	imposta limitazioni alle risorse (ulimit, core, ...) con il modulo pam_limits(8)
/etc/security/time.conf	imposta limitazioni temporali con il modulo pam_time(8)

Tabella 4.6: Elenco di file di configurazione letti da PAM e NSS

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.

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 `glibc-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

L'installazione del pacchetto `libpam-cracklib` permette di imporre regole più stringenti sulle password, per esempio usando righe attive nel file `"/etc/pam.d/common-password"` come le seguenti.

Per squeeze:

```
password required pam_cracklib.so retry=3 minlen=9 difok=3
password [success=1 default=ignore] pam_unix.so use_authtok nullok md5
password requisite pam_deny.so
password required pam_permit.so
```

4.6 Altri controlli sugli accessi

Nota

Vedere Sezione [9.3.15](#) per limitare la funzionalità [SAK \(Secure Attention Key\)](#) del kernel.

Suggerimento

In [systemd](#), si può usare [logind](#) per gestire il login degli utenti. Vedere `systemd-logind(8)`.

4.6.1 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 <mio_nome> <un_certo_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"`.

4.6.2 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)`.

4.6.3 SELinux

[SELinux \(Security-Enhanced Linux\)](#) è un'infrastruttura per rendere il modello basato su privilegi più restrittivo del modello ordinario di sicurezza in stile Unix, usando politiche di [controllo obbligatorio degli accessi \(MAC, mandatory access control\)](#). I poteri di root possono essere limitati in determinate condizioni.

4.6.4 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>"`
- configurazione del runlevel per il [demone](#)
- [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](#)

Vedere Sezione [3.2.3](#), Sezione [3.2.4](#), Sezione [4.5.1](#), Sezione [3.2.8](#) e Sezione [5.9](#).

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 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**.

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

4.7.1 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.

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.)

4.7.2 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 inoltro 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 [6.9](#).

4.7.3 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 [5.9](#)) insieme agli strumenti di sicurezza elencati in seguito, può migliorare la sicurezza generale.

pacchetto	popcon	dimensione	descrizione
knockd	V:0, I:3	89	demone, knockd(1), e client, knock(1) piccoli per bussare alle porte
fail2ban	V:100, I:111	1302	strumento per interdire IP che causano errori di autenticazione multipli
libpam-shield	V:0, I:0	106	blocca all'esterno gli attacchi remoti che cercano di indovinare le password

Tabella 4.8: Elenco di strumenti per fornire misure aggiuntive di sicurezza

4.7.4 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
- Bloccare il BIOS ed impedire l'avvio da supporti removibili
- 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. Spostare il disco fisso in un PC con un BIOS che permette l'avvio da CD
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 [9.4.11](#)). 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 [9.8](#)). Tuttavia è sempre necessaria la password per avviare il sistema.

Capitolo 5

Impostazione della rete

Suggerimento

Per una guida generica all'uso della rete in GNU/Linux, leggere la [Linux Network Administrators Guide](#).

Suggerimento

Per una guida moderna sull'uso della rete specifica per Debian, leggere [The Debian Administrator's Handbook — Configuring the Network](#).

**avvertimento**

Invece di usare lo schema di nomi tradizionale per le interfacce («eth0», «eth1», «wlan0», ...), il più recente [systemd](#) usa «nomi di interfacce di rete prevedibili (Predictable Network Interface Name)», come «enp0s25».

**avvertimento**

Questo capitolo sta diventando non aggiornato dato che è basato su Debian 7.0 (wheezy), rilasciata nel 2013.

Suggerimento

Anche se questo documento usa ancora il vecchio ifconfig(8) con IPv4 per i suoi esempi di configurazione della rete, Debian sta migrando a ip(8) con IPv4+IPv6 nel rilascio wheezy. Patch per aggiornare questo documento sono le benvenute.

Suggerimento

In [systemd](#), si può usare [networkd](#) per gestire 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.

pacchetto	popcon	dimensione	tipo	descrizione
ifupdown	V:591, I:995	199	config::ifupdown	strumento standard per attivare e disattivare la rete (specifico di Debian)
ifplugd	V:4, I:21	214	" "	gestisce automaticamente la rete cablata
ifupdown-extra	V:0, I:1	96	" "	script per testare la rete di supporto al pacchetto "ifupdown"
ifmetric	V:0, I:1	36	" "	imposta la metrica di routing per un'interfaccia di rete
guessnet	V:0, I:0	422	" "	script di mappatura di supporto al pacchetto "ifupdown", attraverso il file "/etc/network/interfaces"
ifscheme	V:0, I:0	58	" "	script di mappatura di supporto al pacchetto "ifupdown"
network-manager	V:373, I:469	10396	config::NM	NetworkManager (demone): gestisce la rete automaticamente
network-manager-gnome	V:183, I:410	5831	" "	NetworkManager (frontend GNOME)
wicd	I:32	35	config::wicd	gestore della rete cablata e wireless (metapacchetto)
wicd-cli	V:0, I:2	59	" "	gestore della rete cablata e wireless (client a riga di comando)
wicd-curses	V:1, I:4	175	" "	gestore della rete cablata e wireless (client Curses)
wicd-daemon	V:27, I:37	954	" "	gestore della rete cablata e wireless (demone)
wicd-gtk	V:22, I:34	574	" "	gestore della rete cablata e wireless (client GTK+)
iptables	V:259, I:995	1517	config::Netfilter	strumenti amministrativi per filtraggio di pacchetti di rete e NAT (Netfilter)
iproute	V:82, I:372	23	config::iproute2	iproute2 , IPv6 e altra configurazione di rete avanzata: ip(8), tc(8), ecc.
ifrename	V:0, I:2	122	" "	rinomina le interfacce di rete in base a vari criteri statici: ifrename(8)
ethtool	V:110, I:267	375	" "	mostra e modifica le impostazioni dei device Ethernet
iputils-ping	V:250, I:996	111	test::iproute2	testa la raggiungibilità di rete di un host remoto con nome host o indirizzo IP (iproute2)
iputils-arping	V:43, I:426	58	" "	testa la raggiungibilità di rete di un host remoto specificato con il suo indirizzo ARP
iputils-tracepath	V:8, I:160	75	" "	traccia il percorso di rete verso un host remoto
net-tools	V:346, I:839	940	config::net-tools	toolkit di rete NET-3 (net-tools , configurazione di rete IPv4): ifconfig(8), ecc.
inetutils-ping	V:0, I:1	336	test::net-tools	testa la raggiungibilità della rete di un host remoto con nome host o indirizzo IP (datato, GNU)
arping	V:2, I:29	72	" "	testa la raggiungibilità di rete di un host remoto specificato con il suo indirizzo ARP (datato)
traceroute	V:66, I:978	154	" "	traccia il percorso di rete verso un host remoto (datato, console)
isc-dhcp-client	V:299, I:968	650	config::low-level	client DHCP
wpa_supplicant	V:294, I:538	2668	" "	supporto client per WPA e WPA2 (IEEE 802.11i)
wpa_gui	V:0, I:3	762	" "	client GUI Qt per WPA supplicant
wireless-tools	V:162, I:276	286	" "	strumenti per manipolare le estensioni wireless per Linux
ppp	V:224, I:513	927	" "	connessione PPP/PPPoE con chat
pppoeconf	V:0, I:10	290	config::helper	strumento di aiuto di configurazione per la connessione PPPoE
pppconfig	V:1, I:3	805	" "	strumento di aiuto per la connessione PPP con chat
wvdial	V:0, I:7	276	" "	strumento di aiuto per la connessione PPP con wvdial e ppp
mtr-tiny	V:6, I:58	140	test::low-level	traccia il percorso di rete verso un host remoto (curses)
mtr	V:5, I:41	190	" "	traccia il percorso di rete verso un host remoto (curses e GTK+)
gnome-nettool	V:5, I:149	2101	" "	strumenti per operazioni comuni sulle informazioni di rete (GNOME)
nmap	V:39, I:340	4673	" "	strumento per mappatura della rete / esplorazione delle porte (Nmap , console)
				strumento per mappatura della rete / esplorazione delle

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 <nome_host>

# Le righe seguenti sono desiderabili per host con supporto per IPv6
::1      ip6-localhost ip6-loopback
fe00::0  ip6-localnet
ff00::0  ip6-mcastprefix
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
ff02::3  ip6-allhosts
```

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

Il nome dell'interfaccia di rete, ad esempio `eth0`, viene assegnato a ciascun hardware, quando viene rilevato, nel kernel Linux attraverso il meccanismo di configurazione in spazio utente `udev` (vedere Sezione [3.3](#)). Il nome dell'interfaccia di rete è chiamato **interfaccia fisica** in `ifup(8)` e `interfaces(5)`.

Per garantire che ciascuna interfaccia di rete abbia un nome persistente ad ogni avvio, usando [indirizzi MAC](#) ecc., esiste un file di regole: `"/etc/udev/rules.d/70-persistent-net.rules"`. Questo file viene generato automaticamente dal programma `"/lib/udev/write_net_rules"` eseguito probabilmente dal file `"persistent-net-generator.rules"`. Lo si può modificare per cambiare le regole di assegnazione dei nomi.



Attenzione

Quando si modifica il file delle regole `"/etc/udev/rules.d/70-persistent-net.rules"`, si deve mantenere ciascuna regola su una singola riga e l'[indirizzo MAC](#) in lettere minuscole. Per esempio, se si trova in questo file una voce `"FireWire device"` e una `"PCI device"`, probabilmente si vorrà chiamare `"PCI device"` `eth0` e configurarla come interfaccia di rete primaria.

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.

Classe	indirizzi di rete	maschera di rete	maschera rete /bit	numero di sottoreti
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 [9.9.6](#).

5.2 La configurazione moderna della rete per il desktop

I sistemi Debian *squeeze* e successivi possono gestire la connessione di rete attraverso software per [demoni](#) di gestione come [NetworkManager \(NM\)](#) (network-manager e pacchetti associati) oppure [Wicd](#) (wicd e pacchetti associati).

- 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"`.

Suggerimento

Con [systemd](#), la rete può essere configurata invece in `/etc/systemd/network/`. Vedere `systemd-resolved(8)`, `resolved.conf(5)` e `systemd-networkd(8)`.

Nota

Alcune funzionalità di questi strumenti automatici di configurazione della rete possono soffrire dei problemi. Non sono così robusti come il vecchio pacchetto `ifupdown`. Controllare il [BTS di network-manager](#) e il [BTS di wicd](#) per vedere i problemi e i limiti attuali.

5.2.1 Strumenti grafici di configurazione della rete

La documentazione ufficiale per NM e Wicd su Debian è fornita rispettivamente in `"/usr/share/doc/network-manager/README"` e `"/usr/share/doc/wicd/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 pippo netdev
```

2. Mantenere la configurazione di `"/etc/network/interfaces"` semplice come la seguente.

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

3. Riavviare NM o Wicd con il comando seguente.

```
$ sudo /etc/init.d/network-manager restart
```

```
$ sudo /etc/init.d/wicd restart
```

4. Configurare la propria rete attraverso l'interfaccia grafica.

Nota

Solo le interfacce che **non** sono elencate in `/etc/network/interfaces` sono gestite da NM o Wicd, per evitare conflitti con `ifupdown`.

Suggerimento

Se si desidera estendere le capacità di configurazione della rete di NM, cercare i moduli plugin appropriati e i pacchetti supplementari, quali `network-manager-openconnect`, `network-manager-openvpn-gnome`, `network-manager-pptp-gnome`, `mobile-broadband-provider-info`, `gnome-bluetooth`, ecc. Lo stesso vale per quelli di Wicd.



Attenzione

Questi strumenti di configurazione automatica della rete possono non essere compatibili con alcune configurazioni esotiche del datato `ifupdown` contenute in `/etc/network/interfaces`, come quelle in Sezione 5.5 e Sezione 5.6. Per informazioni sui problemi e le limitazioni attualmente esistenti, controllare il [BTS di network-manager](#) e il [BTS di wicd](#).

5.3 La vecchia connessione e configurazione di rete

Quando il metodo descritto in Sezione 5.2 non è sufficiente per le proprie necessità, si dovrebbe usare il metodo datato di connessione e configurazione della rete che combina assieme diversi strumenti più semplici.

La connessione di rete in vecchio stile è specifica per ciascun metodo (vedere Sezione 5.4).

Esistono 2 tipi di programma per la configurazione a basso livello della rete in Linux (vedere Sezione 5.7.1).

- I vecchi programmi [net-tools](#) (`ifconfig(8)`, ...) provengono dal sistema di rete Linux NET-3. La maggior parte di essi è ora superata.
- I nuovi programmi [Linux iproute2](#) (`ip(8)`, ...) formano l'attuale sistema Linux di rete.

Anche se questi programmi di rete a basso livello sono potenti, sono complicati da usare. Perciò sono stati creati strumenti di configurazione della rete di alto livello.

Il pacchetto `ifupdown` è lo standard de facto per una configurazione di rete di altro livello di questo tipo in Debian. Permette di attivare la rete semplicemente eseguendo, per esempio, `ifup eth0`. Il suo file di configurazione è il file `/etc/network/interfaces` e i suoi contenuti hanno tipicamente un aspetto simile al seguente.

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp
```

Il pacchetto `resolvconf` è stato creato per fornire al sistema `ifupdown` il supporto per una riconfigurazione indolore della risoluzione degli indirizzi di rete, riscrivendo automaticamente il file `"/etc/resolv.conf"` di configurazione del risolutore. Attualmente la maggior parte dei pacchetti di configurazione della rete in Debian sono modificati per usare il pacchetto `resolvconf` (vedere `"/usr/share/doc/resolvconf/README.Debian"`).

Per automatizzare la configurazione dinamica di ambienti di rete come un PC portatile in una LAN cablata, sono stati creati script di aiuto per il pacchetto `ifupdown`, come `ifplugd`, `guessnet`, `ifscheme`, ecc. Essi sono relativamente difficili da usare ma si comportano bene con il sistema `ifupdown` esistente.

Questi sono spiegati in dettaglio con esempi (vedere Sezione 5.5 e Sezione 5.6).

5.4 Il metodo di connessione alla rete (datato)



Attenzione

I metodi per testare la connessione descritti in questa sezione sono pensati per fare prove. Non sono pensati per essere usati direttamente per connettersi alla rete quotidianamente. È consigliato il loro uso con NM, Wicd o il pacchetto `ifupdown` (vedere Sezione 5.2 e Sezione 5.5).

I metodi di connessione tipici alla rete e i percorsi tipici per la connessione per un PC possono essere schematizzati nel modo seguente.

PC	metodo di connessione	percorso di connessione
Porta seriale (ppp0)	PPP	modem POTS punto di accesso dial-up ISP
Porta ethernet (eth0)	PPPoE/DHCP/Statico	modem BB (banda larga) servizio BB punto di accesso BB ISP
Porta ethernet (eth0)	DHCP/Statico	LAN router BB (banda larga) con NAT (traduzione degli indirizzi di rete) (modem BB ...)

Tabella 5.3: Elenco dei metodi di connessione alla rete e dei percorsi di connessione

Quella che segue è una schematizzazione degli script di configurazione per ciascun metodo di connessione.

metodo di connessione	configurazione	pacchetto(i) backend
PPP	<code>pppconfig</code> per creare chat deterministiche	<code>pppconfig</code> , <code>ppp</code>
PPP (alternativo)	<code>wvdialconf</code> per creare chat euristiche	<code>ppp</code> , <code>wvdial</code>
PPPoE	<code>pppoeconf</code> per creare chat deterministiche	<code>pppoeconf</code> , <code>ppp</code>
DHCP	descritto in <code>"/etc/dhcp/dhclient.conf"</code>	<code>isc-dhcp-client</code>
IP statico (IPv4)	descritto in <code>"/etc/network/interfaces"</code>	<code>iproute</code> o <code>net-tools</code> (obsolete)
IP statico (IPv6)	descritto in <code>"/etc/network/interfaces"</code>	<code>iproute</code>

Tabella 5.4: Elenco di configurazioni della connessione di rete

Gli acronimi per le connessioni di rete hanno il significato seguente.

acronimo	significato
POTS	Plain Old Telephone Service, il semplice vecchio servizio telefonico
BB	banda larga
servizio BB (BB-service)	ad esempio, la linea digitale (DSL), la TV via cavo o fibre ottiche
modem BB	ad esempio, il modem DSL , il modem via cavo o l' ONT , il terminale di rete ottica
LAN	rete locale (Local Area Network)
WAN	rete geografica (Wide Area Network)
DHCP	Protocollo di Configurazione Dinamica dell'Host
PPP	protocollo Point-to-Point
PPPoE	protocollo Point-to-Point via Ethernet
ISP	Fornitore di servizi Internet (Internet Service Provider)

Tabella 5.5: Elenco di acronimi per le connessioni di rete

Nota

I servizi di connessione WAN attraverso la TV via cavo sono generalmente serviti da DHCP o PPPoE. Quelli ADSL e FTTP sono generalmente serviti da PPPoE. Si deve consultare il proprio fornitore di servizi Internet per i requisiti esatti di configurazione della connessione WAN.

Nota

Quando un router BB è usato per creare un ambiente LAN domestico, i PC della LAN sono connessi alla WAN attraverso il router BB con [NAT \(traduzione degli indirizzi di rete\)](#). In questi casi alle interfacce di rete dei PC nella LAN sono assegnati dal router BB indirizzi IP statici o DHCP. Il router BB deve essere configurato per connettersi alla WAN seguendo le istruzioni del proprio fornitore di servizi Internet.

5.4.1 Connessione DHCP con Ethernet

Le reti tipiche moderne per le abitazioni e le piccole aziende, cioè le LAN, sono connesse alla WAN (Internet) usando un router per banda larga economico. La LAN dietro questo router è di solito servita dal server [DHCP](#) (protocollo di configurazione dinamica dell'host) in esecuzione sul router.

Installare semplicemente il pacchetto `isc-dhcp-client` per Ethernet con [DHCP \(protocollo di configurazione dinamica dell'host\)](#).

Vedere `dhclient.conf(5)`.

5.4.2 Connessione con IP statico con Ethernet

Per Ethernet con IP statico non è necessaria alcuna azione particolare.

5.4.3 Connessione PPP con pppconfig

Lo script di configurazione `pppconfig` imposta la connessione [PPP](#) interattivamente chiedendo all'utente di selezionare solamente quanto segue.

- Il numero di telefono
 - Il nome utente per il fornitore di servizi Internet
 - La password per il fornitore di servizi Internet
-

- La velocità della porta
- La porta di comunicazione del modem
- Il metodo di autenticazione

file	funzione
/etc/ppp/peers/<nome_isp>	Il file di configurazione per <code>pppd</code> specifico per il fornitore di servizi <nome_isp>, generato da <code>pppconfig</code>
/etc/chatscripts/<nome_isp>	Il file di configurazione per <code>chat</code> specifico per il fornitore di servizi <nome_isp>, generato da <code>pppconfig</code>
/etc/ppp/options	I parametri generici di esecuzione per <code>ppd</code>
/etc/ppp/pap-secret	Dati di autenticazione per PAP (rischio per la sicurezza)
/etc/ppp/chap-secret	Dati di autenticazione per PAP (più sicuri)

Tabella 5.6: Elenco dei file di configurazione per la connessione [PPP](#) con `pppconfig`



Attenzione

Se i comandi `pon` e `poff` vengono invocati senza argomenti, "<nome_isp>" viene impostato a "provider".

Si può testare la configurazione usando strumenti di configurazione di rete a basso livello come nel modo seguente.

```
$ sudo pon <nome_isp>
...
$ sudo poff <nome_isp>
```

Vedere `"/usr/share/doc/ppp/README.Debian.gz"`.

5.4.4 Connessione PPP alternativa con `wvdialconf`

Un approccio diverso all'uso di `pppd`(8) è la sua esecuzione tramite `wvdial`(1) che è fornito nel pacchetto `"wvdial"`. Invece di essere `pppd` ad eseguire `chat`(8) per comporre il numero e negoziare la connessione, è `wvdial` che fa queste cose e poi avvia `pppd` per tutto il resto.

Lo script di configurazione `wvdialconf` imposta la connessione PPP interattivamente chiedendo all'utente di selezionare solamente quanto segue.

- Il numero di telefono
- Il nome utente per il fornitore di servizi Internet
- La password per il fornitore di servizi Internet

Nella maggior parte dei casi `wvdial` crea la connessione con successo e mantiene automaticamente un elenco dei dati di autenticazione.

Si può testare la configurazione usando strumenti di configurazione di rete a basso livello come nel modo seguente.

```
$ sudo wvdial
...
$ sudo killall wvdial
```

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

file	funzione
/etc/ppp/peers/wvdial	Il file di configurazione per pppd specifico per wvdial, generato da wvdialconf
/etc/wvdial.conf	Il file di configurazione creato da wvdialconf
/etc/ppp/options	I parametri generici di esecuzione per pppd
/etc/ppp/pap-secret	Dati di autenticazione per PAP (rischio per la sicurezza)
/etc/ppp/chap-secret	Dati di autenticazione per PAP (più sicuri)

Tabella 5.7: Elenco dei file di configurazione per la connessione PPP con wvdialconf

5.4.5 Connessione PPPoE con pppoeconf

Quando il fornitore di servizi Internet dà una connessione PPPoE e si decide di connettere il proprio PC direttamente alla WAN, la rete del proprio PC deve essere configurata con PPPoE. PPPoE sta per "PPP over Ethernet" (PPP via Ethernet). Lo script di configurazione `pppoeconf` configura la connessione PPPoE in modo interattivo.

I file di configurazione sono i seguenti.

file	funzione
/etc/ppp/peers/dsl-provider	Il file di configurazione per pppd specifico per pppoe, generato da pppoeconf
/etc/ppp/options	I parametri generici di esecuzione per pppd
/etc/ppp/pap-secret	Dati di autenticazione per PAP (rischio per la sicurezza)
/etc/ppp/chap-secret	Dati di autenticazione per PAP (più sicuri)

Tabella 5.8: Elenco dei file di configurazione per la connessione PPPoE con pppoeconf

Si può testare la configurazione usando strumenti di configurazione di rete a basso livello come nel modo seguente.

```
$ sudo /sbin/ifconfig eth0 up
$ sudo pon dsl-provider
...
$ sudo poff dsl-provider
$ sudo /sbin/ifconfig eth0 down
```

Vedere `"/usr/share/doc/pppoeconf/README.Debian"`.

5.5 Configurazione base della rete con ifupdown (metodo vecchio)

La configurazione tradizionale della rete TCP/IP in un sistema Debian usa il pacchetto `ifupdown` come strumento ad alto livello. Ci sono 2 casi tipici.

- Per sistemi con **IP dinamico**, come PC portatili, si dovrebbe impostare la rete TCP/IP **con** il pacchetto `resolvconf` che permette di cambiare facilmente la configurazione di rete (vedere Sezione 5.5.4).
- Per sistemi con **IP statico**, come server, si dovrebbe impostare la rete TCP/IP **senza** il pacchetto `resolvconf` e mantenere semplice il proprio sistema (vedere Sezione 5.5.5).

Questi metodi di impostazione tradizionali sono piuttosto utili se si desidera impostare una configurazione avanzata; si possono trovare i dettagli qui di seguito.

Il pacchetto `ifupdown` fornisce l'infrastruttura standard per la configurazione di rete di alto livello nel sistema Debian. In questa sezione viene descritta la configurazione base della rete con `ifupdown` con un'introduzione semplificata e molti esempi tipici.

5.5.1 Sintassi di base dei comandi

Il pacchetto `ifupdown` contiene 2 comandi: `ifup(8)` e `ifdown(8)`. Offrono una configurazione della rete di alto livello in base al file di configurazione `"/etc/network/interfaces"`.

comando	azione
<code>ifup eth0</code>	attiva una interfaccia di rete <code>eth0</code> con la configurazione di <code>eth0</code> se esiste un blocco <code>"iface eth0"</code>
<code>ifdown eth0</code>	disattiva una interfaccia di rete <code>eth0</code> con la configurazione di <code>eth0</code> se esiste un blocco <code>"iface eth0"</code>

Tabella 5.9: Elenco di comandi base per la configurazione della rete con `ifupdown`



avvertimento

Non usare strumenti di configurazione a basso livello, come i comandi `ifconfig(8)` e `ip(8)`, per configurare un'interfaccia nello stato **attivo (up)**.

Nota

Non esiste un comando `ifupdown`.

5.5.2 Sintassi di base di `"/etc/network/interfaces"`

La sintassi di base di `"/etc/network/interfaces"`, come spiegato in `interfaces(5)`, può essere riassunta nel modo seguente.

blocco	significato
<code>"auto <nome_interfaccia>"</code>	attiva l'interfaccia <code><nome_interfaccia></code> all'avvio del sistema
<code>"allow-auto <nome_interfaccia>"</code>	<code>" "</code>
<code>"allow-hotplug <nome_interfaccia>"</code>	attiva l'interfaccia <code><nome_interfaccia></code> quando il kernel rileva un evento hotplug dall'interfaccia
Le righe che iniziano con <code>"iface <nome_config> ..."</code>	definiscono la configurazione di rete <code><nome_config></code>
Le righe che iniziano con <code>"mapping <glob_nome_interfaccia>"</code>	definiscono i valori di mappatura di <code><nome_config></code> al corrispondente <code><nome_interfaccia></code>
Le righe che iniziano con un carattere cancellato <code>"#"</code>	sono commenti e vengono ignorate (i commenti di fine riga non sono supportati)
Le righe che terminano con una barra rovesciata <code>"\"</code>	estendono la configurazione alla riga seguente

Tabella 5.10: Elenco dei blocchi in `"/etc/network/interfaces"`

Le righe con un blocco **iface** hanno la sintassi seguente.

```
iface <nome_config> <famiglia_indirizzi> <nome_metodo>
<opzione1> <valore1>
<opzione2> <valore2>
...
```

Per la configurazione di base non vengono usati i blocchi **mapping** ma viene usato il nome dell'interfaccia di rete come nome della configurazione di rete (vedere Sezione [5.6.5](#)).

**avvertimento**

Non creare definizioni duplicate per un blocco "iface" per un'interfaccia di rete in "/etc/network/interfaces".

5.5.3 Interfaccia di rete loopback

La voce di configurazione seguente, nel file "/etc/network/interfaces" attiva l'interfaccia di rete loopback, `lo`, all'avvio del sistema (attraverso il blocco **auto**).

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

Questo blocco esiste sempre nel file "/etc/network/interfaces".

5.5.4 Interfaccia di rete servita da DHCP

Dopo aver preparato il sistema con Sezione 5.4.1, l'interfaccia di rete con DHCP viene configurata creando una voce di configurazione nel file "/etc/network/interfaces" nel modo seguente.

```
allow-hotplug eth0
iface eth0 inet dhcp
```

Quando il kernel Linux rileva l'interfaccia fisica `eth0`, il blocco **allow-hotplug** fa sì che `ifup` attivi l'interfaccia e il blocco **iface** fa sì che `ifup` usi DHCP per configurare l'interfaccia.

5.5.5 Interfaccia di rete con IP statico

L'interfaccia di rete con IP statico viene configurata creando una voce di configurazione nel file "/etc/network/interfaces" nel modo seguente.

```
allow-hotplug eth0
iface eth0 inet static
    address 192.168.11.100
    netmask 255.255.255.0
    gateway 192.168.11.1
    dns-domain example.com
    dns-nameservers 192.168.11.1
```

Quando il kernel Linux rileva l'interfaccia fisica `eth0`, il blocco **allow-hotplug** fa sì che `ifup` attivi l'interfaccia e il blocco **iface** fa sì che `ifup` usi l'IP statico per configurare l'interfaccia.

In questo esempio sono stati dati per veri i seguenti dati.

- Intervallo di indirizzi IP della rete LAN: 192.168.11.0 - 192.168.11.255
 - Indirizzo IP del gateway: 192.168.11.1
 - Indirizzo IP del PC: 192.168.11.100
 - Il pacchetto `resolvconf` è installato.
 - Nome di dominio: "example.com"
 - Indirizzo IP del server DNS: 192.168.11.1
-

Quando il pacchetto `resolvconf` non è installato, la configurazione relativa al DNS deve essere fatta manualmente modificando `"/etc/resolv.conf"` nel modo seguente.

```
nameserver 192.168.11.1
domain example.com
```



Attenzione

L'indirizzo IP usato nell'esempio soprastante non deve essere copiato pari pari. È necessario mettere i numeri IP della reale configurazione della propria rete.

5.5.6 Informazioni di base sull'interfaccia LAN wireless

La [LAN wireless \(in breve WLAN\)](#) fornisce connettività veloce wireless attraverso comunicazioni con espansione di spettro in bande radio senza licenza basate sull'insieme di standard chiamato [IEEE 802.11](#).

Le interfacce WLAN sono quasi come normali interfacce Ethernet, ma richiedono alcuni dati sull'identità della rete e sulla chiave di cifratura per essere usate una volta inizializzate. Gli strumenti di rete di alto livello per esse sono esattamente gli stessi delle interfacce di rete, tranne che i nomi delle interfacce sono un po' diversi come `eth1`, `wlan0`, `ath0`, `wifi0`, ... a seconda dei driver del kernel utilizzati.

Suggerimento

Il device `wmaster0` è il device master che è un device interno usato solamente da [SoftMAC](#) con la nuova [API mac80211 di Linux](#).

Quelle che seguono sono alcune parole chiave relative alla WLAN che è bene ricordare.

acronimo	forma estesa	significato
NWID	Network ID (ID di rete)	identificativo di rete a 16 bit usato da reti WaveLAN pre-802.11 (molto deprecato)
(E)SSID	(Extended) Service Set Identifier (Identificativo di insieme di servizi (esteso))	il nome di rete dei punti di accesso wireless (Wireless Access Points, AP) interconnessi a formare una LAN wireless 802.11 ; identificativo del dominio
WEP, (WEP2)	Wired Equivalent Privacy (Privacy equivalente alla cablata)	standard di cifratura per wireless a 64 bit (128 bit) di prima generazione con chiave a 40-bit (deprecato)
WPA	Wi-Fi Protected Access (Accesso Wi-Fi protetto)	standard di cifratura per wireless di seconda generazione (maggior parte del 802.11i), compatibile con WEP
WPA2	Wi-Fi Protected Access 2 (Accesso Wi-Fi protetto 2)	standard di cifratura per wireless di terza generazione (tutto il 802.11i), non compatibile con WEP

Tabella 5.11: Elenco di acronimi per WLAN

Di fatto la scelta del protocollo è di solito limitata dal router wireless che si possiede.

5.5.7 Interfaccia LAN wireless con WPA/WPA2

È necessario installare il pacchetto `wpa_supplicant` per supportare la WLAN con i nuovi WPA/WPA2.

In caso di IP [DHCP](#) per la connessione WLAN, la voce nel file `"/etc/network/interfaces"` dovrebbe essere come la seguente.


```
allow-hotplug ath0
iface ath0 inet dhcp
wpa-ssid zonacasa
# la chiave di cifratura, psk, esadecimale e' codificata da una passphrase in puro testo
wpa-psk 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f
```

Vedere `"/usr/share/doc/wpa_supplicant/README.modes.gz"`.

5.5.8 Interfaccia LAN wireless con WEP

È necessario installare il pacchetto `wireless-tools` per supportare la rete WLAN con il vecchio WEP. (Il proprio router economico potrebbe usare ancora questa infrastruttura non sicura, ma è meglio di niente.)



Attenzione

Notare che il traffico di rete in una WLAN con WEP può essere spiato da altri.

In caso di IP [DHCP](#) per la connessione WLAN, la voce nel file `"/etc/network/interfaces"` dovrebbe essere come la seguente.

```
allow-hotplug eth0
iface eth0 inet dhcp
wireless-essid Casa
wireless-key1 0123-4567-89ab-cdef
wireless-key2 12345678
wireless-key3 s:password
wireless-defaultkey 2
wireless-keymode open
```

Vedere `"/usr/share/doc/wireless-tools/README.Debian"`.

5.5.9 Connessione PPP

È necessario come prima cosa configurare la connessione PPP come descritto in precedenza (vedere Sezione [5.4.3](#)). Quindi aggiungere nel file `"/etc/network/interfaces"` la voce per il device PPP primario `ppp0` nel modo seguente.

```
iface ppp0 inet ppp
provider <nome_isp>
```

5.5.10 Connessione PPP alternativa

È necessario come prima cosa configurare la connessione PPP alternativa con `wvdial`, come descritto in precedenza (vedere Sezione [5.4.4](#)). Quindi aggiungere nel file `"/etc/network/interfaces"` la voce per il device PPP primario `ppp0` nel modo seguente.

```
iface ppp0 inet wvdial
```

5.5.11 Connessione PPPoE

Per i PC connessi direttamente alla WAN con servizio PPPoE, è necessario configurare il sistema con la connessione PPPoE come descritto in precedenza (vedere Sezione 5.4.5). Quindi aggiungere nel file `/etc/network/interfaces` la voce per il device PPPoE primario `eth0` nel modo seguente.

```
allow-hotplug eth0
iface eth0 inet manual
pre-up /sbin/ifconfig eth0 up
up ifup ppp0=dsl
down ifdown ppp0=dsl
post-down /sbin/ifconfig eth0 down
# Quello che segue e' usato solo internamente
iface dsl inet ppp
provider dsl-provider
```

5.5.12 Lo stato di configurazione della rete di ifupdown

Il file `/etc/network/run/ifstate` memorizza gli stati **previsti** di configurazione della rete per tutte le interfacce di rete attualmente attive gestite dal pacchetto `ifupdown`. Sfortunatamente, anche se il tentativo del sistema `ifupdown` di attivare un'interfaccia come previsto fallisce, il file `/etc/network/run/ifstate` la elenca come attiva.

Se l'output del comando `ifconfig(8)` per un'interfaccia non ha una riga come quella dell'esempio seguente, l'interfaccia non può essere usata come parte di una [rete IPV4](#).

```
inet addr:192.168.11.2 Bcast:192.168.11.255 Mask:255.255.255.0
```

Nota

Per i dispositivi Ethernet connessi con PPPoE, l'output di `ifconfig(8)` manca di una riga come quella dell'esempio precedente.

5.5.13 Riconfigurazione base della rete

Quando si cerca di riconfigurare un'interfaccia, ad esempio `eth0`, per prima cosa è necessario disattivarla con il comando **`sudo ifdown eth0`**. Questo rimuove la voce corrispondente ad `eth0` dal file `/etc/network/run/ifstate`. (Potrebbero essere generati messaggi di errore se `eth0` non è attiva o se non era stata configurata in modo corretto.) Per il momento, fare questa azione per una postazione di lavoro per singolo utente sembra sicuro in qualunque momento.

Si può ora modificare liberamente il contenuto di `/etc/network/interfaces` in base alle proprie necessità per riconfigurare l'interfaccia di rete `eth0`.

In seguito si può riattivare `eth0` con il comando **`sudo ifup eth0`**.

Suggerimento

Si può (re)inizializzare l'interfaccia di rete semplicemente usando **`sudo ifdown eth0;sudo ifup eth0`**.

5.5.14 Pacchetto ifupdown-extra

Il pacchetto `ifupdown-extra` fornisce facili test della connessione di rete per l'uso con il pacchetto `ifupdown`.

- Il comando `network-test(1)` può essere usato dalla shell.
 - Gli script automatici vengono eseguiti per ogni invocazione del comando `ifup`.
-

Il comando `network-test` libera l'utente dal faticoso compito di eseguire comandi a basso livello per analizzare i problemi di rete.

Gli script automatici vengono installati in `"/etc/network/*/"` e effettuano le azioni seguenti.

- Verifica della connessione del cavo di rete
- Verifica dell'eventuale uso duplicato di indirizzi IP
- Impostazione di instradamenti statici del sistema in base alle definizioni in `"/etc/network/routes"`
- Verifica della raggiungibilità del gateway di rete
- Registrazione dei risultati nel file `"/var/log/syslog"`

Queste voci nel syslog sono piuttosto utili per l'amministrazione di problemi di rete nel sistema remoto.

Suggerimento

Il comportamento automatico del pacchetto `ifupdown-extra` è configurabile con `"/etc/default/network-test"`. Alcuni di questi controlli automatici rallentano un po' il sistema all'avvio dato che è necessario un po' di tempo per restare in ascolto per le risposte [ARP](#).

5.6 Configurazione avanzata della rete con ifupdown (strumento datato)

Le funzionalità del pacchetto `ifupdown` possono essere migliorate andando oltre quanto è stato descritto in Sezione [5.5](#), con qualche informazione più avanzata.

Le funzionalità descritte in questa sezione sono completamente opzionali. Essendo io pigro e minimalista, raramente faccio lo sforzo di usarle.



Attenzione

Se non si è riusciti a configurare la connessione di rete usando le informazioni in Sezione [5.5](#), cercare una soluzione nelle informazioni che seguono non farà che peggiorare le cose.

5.6.1 Il pacchetto ifplugd

Il pacchetto `ifplugd` è un vecchio strumento di configurazione della rete che può gestire soltanto connessioni Ethernet. Risolve problemi di cavi Ethernet sconnessi/riconnessi per PC portatili, ecc. Se è installato [NetworkManager](#) o [Wicd](#) (vedere Sezione [5.2](#)), questo pacchetto non è necessario.

Questo pacchetto esegue un [demone](#) e rimpiazza le funzionalità `auto` o `allow-hotplug` (vedere Tabella [5.10](#)) ed abilita le interfacce quando queste vengono connesse alla rete.

Ecco come usare il pacchetto `ifplugd` per la porta Ethernet interna, ad esempio `eth0`.

1. Rimuovere il blocco `"auto eth0"` o `"allow-hotplug eth0"` dal file `"/etc/network/interfaces"`.
 2. Mantenere i blocchi `"iface eth0 inet ..."` e `"mapping ..."` nel file `"/etc/network/interfaces"`.
 3. Installare il pacchetto `ifplugd`.
 4. Eseguire `"sudo dpkg-reconfigure ifplugd"`.
 5. Indicare `eth0` come "interfaccia statica che ifplugd deve osservare".
-

Ora, dopo la riconfigurazione, la rete funziona come voluto.

- All'avvio o al rilevamento dell'hardware, l'interfaccia non viene attivata da sé stessa.
 - Procedura di avvio veloce senza lunghi tempi di attesa per DHCP.
 - Nessuna interfaccia attivata in modo inconsueto senza un appropriato indirizzo IPv4 (vedere Sezione [5.5.12](#)).
- Quando viene rilevato il cavo Ethernet l'interfaccia viene attivata.
- Dopo un certo intervallo di tempo dalla disconnessione del cavo Ethernet, l'interfaccia viene automaticamente disattivata.
- Quando viene inserito un altro cavo Ethernet, l'interfaccia è attivata nel nuovo ambiente.

Suggerimento

Gli argomenti del comando `ifplugd(8)` possono impostare il suo comportamento, come il ritardo nella riconfigurazione delle interfacce.

5.6.2 Il pacchetto `ifmetric`

Il pacchetto `ifmetric` permette di manipolare la metrica di instradamenti a posteriori, anche per DHCP.

Le azioni seguenti impostano l'interfaccia `eth0` come preferita rispetto alla interfaccia `wlan0`.

1. Installare il pacchetto `ifmetric`.
2. Aggiungere una riga con l'opzione `"metric 0"` immediatamente sotto alla riga `"iface eth0 inet dhcp"` in `" /etc/network/interfaces"`.
3. Aggiungere una riga con l'opzione `"metric 1"` immediatamente sotto alla riga `"iface wlan0 inet dhcp"` in `" /etc/network/interfaces"`.

`"metric 0"` significa instradamento con la più alta priorità ed è quello predefinito. Un valore di metrica più alto indica instradamenti con priorità più bassa. L'indirizzo IP dell'interfaccia attiva con il valore di metrica più basso diventa quello di origine. Vedere `ifmetric(8)`.

5.6.3 L'interfaccia virtuale

Una singola interfaccia fisica Ethernet può essere configurata come interfacce virtuali multiple con indirizzi IP differenti. Di solito ciò ha lo scopo di connettere un'interfaccia a diverse sottoreti di IP. Per esempio, un'applicazione di questo tipo è l'hosting web virtuale basato su indirizzi IP con una singola interfaccia di rete.

Per esempio, si supponga quanto segue.

- Una singola interfaccia Ethernet dell'host è connessa ad un hub Ethernet (non al router a banda larga).
- L'hub Ethernet è connesso sia ad Internet sia alla rete LAN.
- La rete LAN usa la sottorete `192.168.0.x/24`.
- L'host usa per Internet indirizzi IP DHCP sull'interfaccia fisica `eth0`.
- L'host usa `192.168.0.1` con l'interfaccia virtuale `eth0:0` per la LAN.

I blocchi seguenti in `" /etc/network/interfaces"` configurano la rete.

```
iface eth0 inet dhcp
metric 0
iface eth0:0 inet static
address 192.168.0.1
netmask 255.255.255.0
network 192.168.0.0
metric 1
```



Attenzione
Anche se questo esempio di configurazione con [traduzione degli indirizzi di rete \(NAT\)](#) usando [netfilter/iptables](#) (vedere Sezione 5.9) può fornire un router a basso prezzo per la LAN usando una sola interfaccia, non c'è alcuna reale funzionalità di firewall con una configurazione di questo tipo. Si dovrebbero usare 2 interfacce fisiche con NAT per rendere sicura la rete locale rispetto ad Internet.

5.6.4 Sintassi avanzata dei comandi

Il pacchetto `ifupdown` offre funzionalità di configurazione avanzata della rete usando il nome di **configurazione della rete** e il nome dell'**interfaccia di rete**. Viene qui usata una terminologia leggermente diversa da quella usata in `ifup(8)` e `interfaces(5)`.

terminologia delle pagine man	terminologia di questo documento	esempi nel testo seguente	descrizione
nome dell' interfaccia fisica	nome dell' interfaccia di rete	<code>lo</code> , <code>eth0</code> , <nome_interfaccia>	nome dato dal kernel Linux (usando il meccanismo <code>udev</code>)
nome dell' interfaccia logica	nome della configurazione di rete	<code>config1</code> , <code>config2</code> , <nome_config>	elemento indicante il nome che segue la parola iface nel file <code>"/etc/network/interfaces"</code>

Tabella 5.12: Elenco della terminologia per i device di rete

I comandi base per la configurazione della rete descritti in Sezione 5.5.1 richiedono che, nel file `"/etc/network/interfaces"`, il nome della **configurazione di rete** indicato nel blocco **iface** corrisponda al nome dell'**interfaccia di rete**.

I comandi avanzati per la configurazione della rete permettono, nel file `"/etc/network/interfaces"`, la distinzione tra il nome della **configurazione di rete** e della **interfaccia di rete**

comando	azione
<code>ifup eth0=config1</code>	attiva una interfaccia di rete <code>eth0</code> con la configurazione <code>config1</code>
<code>ifdown eth0=config1</code>	disattiva una interfaccia di rete <code>eth0</code> con la configurazione <code>config1</code>
<code>ifup eth0</code>	attiva una interfaccia di rete <code>eth0</code> con la configurazione scelta dal blocco di mappatura mapping
<code>ifdown eth0</code>	disattiva una interfaccia di rete <code>eth0</code> con la configurazione scelta dal blocco di mappatura mapping

Tabella 5.13: Elenco di comandi avanzati per la configurazione della rete con `ifupdown`

5.6.5 Il blocco di mappatura "mapping"

Nella sezione Sezione 5.5.2 è stata saltata la spiegazione del blocco **mapping** nel file `"/etc/network/interfaces"`, per evitare complicazioni. Questo blocco ha la sintassi seguente.

```
mapping <glob_nome_interfaccia>
    script <nome_script>
    map <script_input1>
    map <script_input2>
    map ...
```

Ciò fornisce funzionalità avanzate al file `/etc/network/interfaces` automatizzando la scelta della configurazione con lo script di mappatura specificato da `<nome_script>`.

Ecco cosa succede quando viene eseguito il comando seguente.

```
$ sudo ifup eth0
```

Quando il `<glob_nome_interfaccia>` trova corrispondenza con `eth0`, l'esecuzione di questo comando produce, per configurare automaticamente `eth0`, l'invocazione del comando seguente.

```
$ sudo ifup eth0=$(echo -e '<script_input1> \n <script_input2> \n ...' | <nome_script> eth0 <
)
```

Le righe, per l'input dello script, che iniziano con `"map"` sono opzionali e possono essere ripetute.

Nota

L'espressione glob per il blocco **mapping** funziona come i metacaratteri dei glob per nomi di file della shell (vedere Sezione [1.5.6](#)).

5.6.6 Configurazione della rete commutabile manualmente

Quella che segue è una descrizione di come commutare manualmente tra diverse configurazioni di rete senza dover riscrivere il file `/etc/network/interfaces` come in Sezione [5.5.13](#).

Per tutte le configurazioni di rete alle quali si deve accedere è sufficiente creare una sezione distinta nel file `/etc/network/interfaces` come quello seguente.

```
auto lo
iface lo inet loopback

iface config1 inet dhcp

iface config2 inet static
    address 192.168.11.100
    netmask 255.255.255.0
    gateway 192.168.11.1
    dns-domain example.com
    dns-nameservers 192.168.11.1

iface pppoe inet manual
    pre-up /sbin/ifconfig eth0 up
    up ifup ppp0=dsl
    down ifdown ppp0=dsl
    post-down /sbin/ifconfig eth0 down

# Quanto segue viene usato solo internamente
iface dsl inet ppp
    provider dsl-provider

iface pots inet ppp
    provider provider
```

Notare che il **nome di configurazione della rete**, che è l'elemento che viene dopo **iface**, non usa il **nome dell'interfaccia di rete**. Inoltre non ci sono blocchi **auto**, né **allow-hotplug** per attivare l'interfaccia di rete **eth0** automaticamente in base ad eventi.

Ora tutto è pronto per commutare tra le configurazioni di rete.

Spostare il PC in una LAN con DHCP. L'**interfaccia di rete** (l'interfaccia fisica) **eth0** viene attivata assegnandole il nome di **configurazione della rete** (il nome dell'interfaccia logica) **config1** nel modo seguente.

```
$ sudo ifup eth0=config1
Password:
...
```

L'interfaccia **eth0** è attiva, configurata con DHCP e connessa alla LAN.

```
$ sudo ifdown eth0=config1
...
```

L'interfaccia di rete **eth0** è disattivata e disconnessa dalla LAN.

Spostare il PC in una LAN con IP statico. L'**interfaccia di rete** **eth0** viene attivata assegnandole il nome di **configurazione della rete** **config2** nel modo seguente.

```
$ sudo ifup eth0=config2
...
```

L'interfaccia **eth0** è attiva, configurata con IP statico e connessa alla LAN. I parametri aggiuntivi **dns-*** forniti configurano il contenuto di **/etc/resolv.conf**. Tale file **/etc/resolv.conf** viene gestito meglio se è installato il pacchetto **resolvconf**.

```
$ sudo ifdown eth0=config2
...
```

L'interfaccia di rete **eth0** è di nuovo disattivata e disconnessa dalla LAN.

Spostare il PC su una porta di un modem per banda larga connesso ad un servizio PPPoE. L'**interfaccia di rete** **eth0** viene attivata assegnandole il nome di **configurazione della rete** **pppoe** nel modo seguente.

```
$ sudo ifup eth0=pppoe
...
```

L'interfaccia **eth0** è attiva, configurata per la connessione PPPoE diretta con il fornitore di servizi Internet.

```
$ sudo ifdown eth0=pppoe
...
```

L'interfaccia di rete **eth0** è di nuovo disattivata e disconnessa.

Spostare il PC in una postazione senza LAN o modem per banda larga, ma con una linea telefonica ed un modem. L'**interfaccia di rete** **ppp0** viene attivata assegnandole il nome di **configurazione della rete** **pots** nel modo seguente.

```
$ sudo ifup ppp0=pots
...
```

L'interfaccia **ppp0** è attiva e connessa ad Internet con PPP.

```
$ sudo ifdown ppp0=pots
...
```

L'interfaccia di rete **ppp0** è disattivata e disconnessa da Internet.

Si dovrebbe controllare nel file **/etc/network/run/ifstate** lo stato attuale della configurazione di rete del sistema **ifupdown**.



avvertimento

Potrebbe essere necessario cambiare i numeri alla fine di `eth*`, `ppp*`, ecc., se si ha più di un'interfaccia di rete.

5.6.7 Uso di script con il sistema ifupdown

Il sistema `ifupdown` esegue automaticamente gli script installati in `"/etc/network/*/"` esportando le variabili d'ambiente negli script.

variabile d'ambiente	valore passato
"\$IFACE"	nome fisico (nome dell'interfaccia) dell'interfaccia in fase di elaborazione
"\$LOGICAL"	nome logico (nome dell'interfaccia) dell'interfaccia in fase di elaborazione
"\$ADDRFAM"	<famiglia_indirizzi> dell'interfaccia
"\$METHOD"	<nome_metodo> dell'interfaccia (es., "static")
"\$MODE"	"start" se invocato da <code>ifup</code> , "stop" se invocato da <code>ifdown</code>
"\$PHASE"	come per "\$MODE", ma in modo più dettagliato, distinguendo le fasi <code>pre-up</code> , <code>post-up</code> , <code>pre-down</code> e <code>post-down</code>
"\$VERBOSITY"	indica se è stato usato <code>"- -verbose"</code> ; se lo è stato viene impostata a 1, altrimenti a 0
"\$PATH"	percorso di ricerca dei comandi: <code>"/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"</code>
"\$IF_<OPZIONE>"	valore della opzione corrispondente all'interno del blocco iface

Tabella 5.14: Elenco delle variabili d'ambiente passate dal sistema `ifupdown`

In questo caso, ciascuna variabile d'ambiente di tipo `"$IF_<OPZIONE>"` viene creata a partire dal nome dell'opzione corrispondente, come `<opzione1>` e `<opzione2>`, preceduto da `"$IF_"` e convertendo tutte le lettere in maiuscolo, sostituendo i caratteri di sottolineatura con trattini e scartando i caratteri non alfanumerici.

Suggerimento

Vedere Sezione 5.5.2 per `<famiglia_indirizzi>`, `<nome_metodo>`, `<opzione1>` e `<opzione2>`.

Il pacchetto `ifupdown-extra` (vedere Sezione 5.5.14) usa queste variabili d'ambiente per estendere la funzionalità del pacchetto `ifupdown`. Il pacchetto `ifmetric` (vedere Sezione 5.6.2) installa lo script `"/etc/network/if-up.d/ifmetric"` che imposta la metrica attraverso la variabile `"$IF_METRIC"`. Anche il pacchetto `guessnet` (vedere Sezione 5.6.8), che fornisce un'infrastruttura semplice e potente per la selezione automatica della configurazione di rete tramite il meccanismo di mappatura, usa queste variabili.

Nota

Per esempi più dettagliati di script personalizzati per la configurazione della rete che usano queste variabili d'ambiente, controllare gli script d'esempio in `"/usr/share/doc/ifupdown/examples/*"` e gli script usati nei pacchetti `ifscheme` e `ifupdown-scripts-zg2`. Questi script aggiuntivi hanno alcune funzionalità che si sovrappongono in parte a quelle dei pacchetti base `ifupdown-extra` e `guessnet`. Se si installano questi script aggiuntivi, si dovrebbe personalizzarli in modo da evitare interferenze.

5.6.8 Mappatura con guessnet

Invece di scegliere manualmente la configurazione, come descritto in Sezione 5.6.6, si può usare il meccanismo di mappatura descritto in Sezione 5.6.5 per selezionare automaticamente la configurazione di rete con script personalizzati.

Il comando `guessnet-ifupdown(8)` fornito nel pacchetto `guessnet` è pensato per essere usato come uno script di mappatura e fornisce un’infrastruttura potente per migliorare il sistema `ifupdown`.

- Per ciascuna configurazione di rete in un blocco **iface**, si elencano delle condizioni da testare come valore per le opzioni di **guessnet**.
- La mappatura sceglie come configurazione di rete la prima **iface** che dà come risultato un non-ERRORE.

Questo doppio uso del file `"/etc/network/interfaces"` da parte dello script di mappatura, `guessnet-ifupdown`, e dell’infrastruttura originale di configurazione della rete, `ifupdown`, non causa impatti negativi dato che le opzioni **guessnet** esportano solo variabili d’ambiente extra a script eseguiti dal sistema `ifupdown`. Vedere i dettagli in `guessnet-ifupdown(8)`.

Nota
Quando in `"/etc/network/interfaces"` è necessaria più di una riga di opzione **guessnet**, usare righe di opzione che iniziano con **guessnet1**, **guessnet2** e così via, dato che il pacchetto `ifupdown` non permette ripetizioni della stringa iniziale delle righe di opzione.

5.7 Configurazione della rete a basso livello

5.7.1 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 5.15: Tabella di traduzione dai comandi obsoleti `net-tools` ai nuovi comandi `iproute2`

Vedere `ip(8)` e [IPROUTE2 Utility Suite Howto](#).

5.7.2 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.

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"`.

comando	descrizione
ifconfig	mostra lo stato del collegamento e l'indirizzo delle interfacce attive
ip addr show	mostra lo stato del collegamento e l'indirizzo delle interfacce attive
route -n	mostra tutta la tabella di instradamento in indirizzi numerici
ip route show	mostra tutta la tabella di instradamento in indirizzi numerici
arp	mostra l'attuale contenuto delle tabelle cache ARP
ip neigh	mostra l'attuale contenuto delle tabelle cache ARP
plog	mostra il registro del demone ppp
ping yahoo.com	controlla la connessione Internet verso "yahoo.com"
whois yahoo.com	controlla chi ha registrato "yahoo.com" nel database dei domini
traceroute yahoo.com	traccia la connessione Internet verso "yahoo.com"
tracpath yahoo.com	traccia la connessione Internet verso "yahoo.com"
mtr yahoo.com	traccia la connessione Internet verso "yahoo.com" (ripetutamente)
dig [@dns-server.com] esempio.com [{a mx any}]	controlla i record DNS di "esempio.com" con "dns-server.com" alla ricerca di un record "a", "mx" o "any"
iptables -L -n	controlla il filtraggio dei pacchetti
netstat -a	trova tutte le porte aperte
netstat -l --inet	trova le porte in ascolto
netstat -ln --tcp	trova le porte (numeriche) TCP in ascolto
dlint esempio.com	controlla le informazioni DNS di zona per "esempio.com"

Tabella 5.16: Elenco di comandi di rete a basso livello

pacchetto	popcon	dimensione	descrizione
iftop	V:8, I:113	93	mostra informazioni sull'uso della banda su di un'interfaccia di rete
iperf	V:4, I:56	168	strumento di misurazione della banda del protocollo Internet
ifstat	V:0, I:10	55	Monitoraggio delle STATistiche sulle InterFacce
bmon	V:1, I:13	140	strumento portatile per monitoraggio della banda e stime della velocità
ethstatus	V:0, I:4	39	script che misura velocemente la quantità di dati passanti per un device di rete
bing	V:0, I:2	71	tester empirico stocastico della banda
bwm-ng	V:2, I:17	89	strumento piccolo e semplice basato sulla console per il monitoraggio della banda
ethstats	V:0, I:1	20	strumento basato su console per il monitoraggio delle statistiche Ethernet
ipfm	V:0, I:0	78	strumento di analisi della banda

Tabella 5.17: Elenco degli strumenti di ottimizzazione della rete

5.8 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.

5.8.1 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
```

Provare 1454 invece di 1500

Come si può vedere `ping(8)` ha avuto successo con 1454.

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)	vedere Configurazione ottimale dell'MTU per connessioni ADSL con PPPoE per il ragionamento

Tabella 5.18: 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.

5.8.2 Impostazione dell'MTU

Quelli che seguono sono alcuni esempi di come impostare il valore di MTU dal suo valore predefinito 1500 a 1454.

Per DHCP (vedere Sezione 5.5.4), si possono sostituire le righe pertinenti nei blocchi **iface** nel file `/etc/network/interfaces` con blocchi come il seguente.

```
iface eth0 inet dhcp
pre-up /sbin/ifconfig $IFACE mtu 1454
```

Per IP statico (vedere Sezione 5.5.5), si possono sostituire le righe pertinenti nei blocchi **iface** nel file `/etc/network/interfaces` con blocchi come il seguente.

```
iface eth0 inet static
address 192.168.11.100
netmask 255.255.255.0
gateway 192.168.11.1
mtu 1454
dns-domain example.com
dns-nameservers 192.168.11.1
```

Per la connessione PPPoE diretta (vedere Sezione 5.4.5, si possono sostituire le riga `mtu` pertinente nel file `/etc/ppp/peers/dsl` con la seguente.

```
mtu 1454
```

Il valore di **MSS (Maximum Segment Size, dimensione massima di segmento)** 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 5.9) può bloccare la dimensione dei pacchetti al valore MSS ed è utile per il router. Vedere `TCPMSS` in `iptables(8)`.

5.8.3 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.

5.9 Infrastruttura netfilter

Netfilter fornisce un'infrastruttura per **firewall con stati** e **traduzione di indirizzi di rete (NAT)** con moduli del **kernel Linux** (vedere Sezione 3.3.1).

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:259, I:995	1517	strumenti di amministrazione per netfilter (iptables(8) per IPv4, ip6tables(8) per IPv6)
arptables	V:0, I:2	84	strumenti di amministrazione per netfilter (arptables(8) per ARP)
ebtables	V:37, I:60	351	strumenti di amministrazione per netfilter (ebtables(8) per bridge Ethernet)
iptstate	V:0, I:5	116	monitors continuamente lo stato di netfilter (in modo simile a top(1))
shorewall-init	V:0, I:0	76	inizializzazione di Shoreline Firewall
shorewall	V:6, I:15	2429	generatore di file di configurazione per Shoreline Firewall , netfilter
shorewall-lite	V:0, I:0	75	generatore di file di configurazione per Shoreline Firewall , netfilter (versione leggera)
shorewall6	V:0, I:2	778	generatore di file di configurazione per Shoreline Firewall , netfilter (versione IPv6)
shorewall6-lite	V:0, I:0	73	generatore di file di configurazione per Shoreline Firewall , netfilter (versione IPv6 leggera)

Tabella 5.19: 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](#).



avvertimento

Questo capitolo sta diventando non aggiornato dato che è basato su Debian 7.0 (wheezy), rilasciata nel 2013.

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:73, I:153	131494	per X	Chromium , (browser open-source da Google)
firefox	V:17, I:31	155475	” ”	Firefox , (browser open-source da Mozilla, disponibile solo in Debian Unstable)
firefox-esr	V:278, I:441	106906	” ”	Firefox ESR , (Firefox Extended Support Release)
iceweasel	V:130, I:275	234	” ”	Mozilla Firefox senza marchio (rimpiazzato da firefox a partire da Debian 9)
epiphany-browser	V:7, I:34	1480	” ”	Epiphany , per GNOME , aderente alle HIG
konqueror	V:23, I:111	20672	” ”	Konqueror , per KDE
dillo	V:0, I:5	1500	” ”	Dillo , (browser leggero, basato su FLTK)
w3m	V:184, I:605	2301	testuali	w3m
lynx	V:30, I:107	1903	” ”	Lynx
elinks	V:17, I:31	1583	” ”	ELinks
links	V:20, I:45	2123	” ”	Links (solo testo)
links2	V:2, I:17	5366	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
icedtea-plugin	I:23	19	main	Plugin Java basato su OpenJDK e IcedTea
mozilla-plugin-gnash	I:1	39	main	Plugin Flash basato su Gnash
flashplugin-nonfree	V:5, I:98	71	contrib	Strumento di aiuto per plugin Flash, per installare Adobe Flash Player (solì i386 e amd64)
browser-plugin-vlc	V:4, I:16	202	main	Plugin multimediale basato sul riproduttore multimediale VLC

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/e" 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



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.

Il sistema di posta comprende molti programmi server e molti programmi client in esecuzione su più host. In base alla funzionalità esistono 3 tipi di programmi server ("agent") per la posta:

- Il Mail Transport Agent ("Agente di trasporto della posta", [MTA](#), vedere Sezione [6.3](#)) è un programma per trasferire la posta tra i diversi host.
- Il Mail Delivery Agent ("Agente di consegna della posta", [MDA](#), vedere Sezione [6.6](#)) è un programma per consegnare i messaggi alle caselle di posta degli utenti all'interno di un host.
- Il Mail User Agent (Agente di posta dell'utente, MUA, chiamato anche [client o programma di posta](#), vedere Sezione [6.4](#)) è il programma per generare messaggi e per accedere ai messaggi consegnati.

Nota

Gli esempi di configurazione che seguono sono validi solamente per la postazione di lavoro mobile su connessioni Internet di largo consumo.

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.)

Il [programma di posta](#) (MUA) deve interpretare i dati nell'intestazione e nel corpo del messaggio usando [MIME \(Multipurpose Internet Mail Extensions\)](#) per gestire il tipo di dati e la codifica del contenuto.

6.2.2 Nozioni di base sui servizi di posta moderni

Per minimizzare l'esposizione allo spam (posta elettronica indesiderata e non richiesta), molti fornitori di servizi Internet che forniscono connessioni Internet domestiche stanno implementando contromisure.

- Il servizio smarthost per i loro clienti per inviare i messaggi usa la porta di inoltro dei messaggi (587) specificata in [rfc4409](#), con la password (servizio [SMTP AUTH](#)) come specificata in [rfc4954](#).
- Le connessioni alla porta (25) [SMTP](#) dagli host interni della loro rete (tranne che per il server della posta in uscita del fornitore di servizi stesso) verso Internet sono bloccate.
- Le connessioni alla porta (25) [SMTP](#) del server di posta in entrata del fornitore di servizi Internet da host di rete esterni sospetti sono bloccate. (La connessione da host su intervalli di indirizzi IP dinamici usati dalle connessioni Internet dial-up e da altre connessioni di tipo domestico sono le prime ad essere bloccate).
- [Tecniche anti-spam](#) come [DKIM \(DomainKeys Identified Mail\)](#) e [SPF \(Sender Policy Framework\)](#) 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.

Queste nuove limitazioni vanno tenute a mente quando si configura il proprio server di posta o si cerca di risolvere problemi di consegna della posta.

Alla luce di queste limitazioni e situazioni ostili in Internet, alcuni fornitori di servizi di posta via Internet indipendenti, come Yahoo.com e Gmail.com, offrono un servizio di posta sicuro a cui ci si può connettere da qualsiasi parte di Internet usando [TLS \(Transport Layer Security\)](#) e il suo predecessore [SSL \(Secure Sockets Layer\)](#).

- Il servizio smarthost sulla porta 465 con il deprecato SMTP via SSL (protocollo [SMTPS](#)).
-

- Il servizio smarthost sulla porta 587 con [STARTTLS](#).
- La posta in entrata è accessibile sulla porta TLS/POP3 (995) con [POP3](#).

**Attenzione**

Avere in esecuzione un server SMTP su una rete ad uso personale per inviare posta direttamente ad host remoti in modo affidabile non è un'ipotesi realistica. È molto probabile che la posta venga rigettata. Si deve usare un qualche servizio smarthost offerto dal proprio fornitore di servizi Internet o da fornitori di servizi di posta via Internet indipendenti.

Per semplicità, nel testo che segue si assume che lo smarthost sia alla posizione "smtp.hostname.dom", che richieda [SMTP AUTH](#) e che usi la porta di invio dei messaggi (587) con [STARTTLS](#).

6.2.3 Strategia di configurazione della posta per postazione PC

La configurazione di posta più semplice possibile è che la posta sia inviata allo smarthost del fornitore di servizi Internet e ricevuta dal server POP3 del fornitore di servizi dal programma di posta (vedere Sezione [6.4](#)) stesso. Questo tipo di configurazione è popolare per i programmi di posta completi basati su interfaccia utente grafica come icedove(1), evolution(1), ecc. Se si necessita di filtrare la posta in base alla sua tipologia, si usano le funzioni di filtro del programma di posta. In questo caso l'MTA (vedere Sezione [6.3](#)) locale deve solo fare la consegna locale della posta (quando il mittente e il destinatario sono sullo stesso host).

Notare che il sistema Debian è un sistema multiutente. Anche se si è l'unico utente, ci sono molti programmi che vengono eseguiti come utente root e che possono inviare un messaggio di posta.

La configurazione di posta alternativa prevede che la posta sia inviata attraverso l'MTA locale allo smarthost del fornitore di servizi Internet e ricevuta dal server POP3 del fornitore di servizi dal programma di recupero della posta (vedere Sezione [6.5](#)) che la mette nella casella di posta locale. Se si necessita di filtrare la posta in base alla sua tipologia, si usa un MDA con filtri (vedere Sezione [6.6](#)) per smistare la posta in caselle diverse. Questo tipo di configurazione è popolare con i semplici programmi di posta basati sulla console come mutt(1), mew(1), ecc., anche se è possibile con qualsiasi programma di posta (vedere Sezione [6.4](#)). In questo caso l'MTA (vedere Sezione [6.3](#)) locale deve fare sia la consegna allo smarthost sia quella locale. Dato che le postazioni di lavoro mobile non hanno un FQDN valido, è necessario configurare l'MTA locale in modo che nasconda e falsifichi il vero nome locale del sistema di posta nei messaggi di posta in uscita per evitare errori nella consegna della posta (vedere Sezione [6.3.3](#)).

Suggerimento

Si potrebbe voler configurare i MUA e MDA in modo che usino il formato [Maildir](#) per archiviare i messaggi di posta elettronica da qualche parte nella propria directory home.

6.3 Agente di trasporto della posta (MTA)

Per le normali postazioni PC, la scelta più comune per il server di posta (MTA) sono i pacchetti `exim4-*` o `postfix`. La scelta spetta all'utente.

Anche dal conteggio dei voti `popcon` `exim4-*` sembra svariare volte più popolare di `postfix`, ciò non significa che `postfix` non sia popolare tra gli sviluppatori Debian. Il sistema server Debian usa sia `exim4` sia `postfix`. Anche l'[analisi delle intestazioni della posta](#) sui messaggi delle mailing list, da parte di sviluppatori Debian di spicco, indica che i due MTA sono ugualmente popolari.

I pacchetti `exim4-*` sono conosciuti per il loro consumo di memoria molto basso e per essere molto flessibili nella configurazione. Il pacchetto `postfix` è conosciuto per essere compatto, veloce, facile e sicuro. Entrambi sono forniti con un'ampia documentazione, sono di alta qualità ed hanno una buona licenza.

Nell'archivio Debian ci sono molte scelte per i pacchetti con agenti di trasporto della posta (MTA) con capacità e scopi diversi.

pacchetto	popcon	dimensione	descrizione
exim4-daemon-light	V:509, I:551	1288	agente di trasporto della posta Exim4 (MTA predefinito in Debian)
exim4-base	V:518, I:564	1582	documentazione (formato testo) e file comuni per Exim4
exim4-doc-html	I:1	3477	documentazione per Exim4 (in formato HTML)
exim4-doc-info	I:1	592	documentazione per Exim4 (in formato info)
postfix	V:149, I:165	3947	agente di trasferimento della posta Postfix (MTA alternativo)
postfix-doc	I:10	4064	documentazione per Postfix (formati HTML e testo)
saslm2-bin	V:7, I:23	410	implementazione dell'API SASL Cyrus (di supporto a Postfix per SMTP AUTH)
cyrus-sasl2-doc	I:2	247	SASL Cyrus - documentazione

Tabella 6.3: Elenco di pacchetti base relativi ai server di trasporto della posta per postazioni PC

pacchetto	popcon	dimensione	capacità e scopo
exim4-daemon-light	V:509, I:551	1288	completa
postfix	V:149, I:165	3947	completa (sicurezza)
exim4-daemon-heavy	V:9, I:10	1429	completa (flessibilità)
sendmail-bin	V:12, I:13	1810	completa (solo se si ha già familiarità con esso)
nullmailer	V:5, I:8	423	ridotta, niente posta locale
ssmtp	V:13, I:19	2	ridotta, niente posta locale
courier-mta	V:0, I:0	2077	molto completa (interfaccia web, ecc.)
masqmail	V:0, I:0	337	leggera
esmtp	V:0, I:0	124	leggera
esmtp-run	V:0, I:0	32	leggera (estensioni per la compatibilità con sendmail per esmtp)
msmtp	V:2, I:7	260	leggera
msmtp-mta	V:1, I:1	35	leggera (estensioni per la compatibilità con sendmail per msmtp)

Tabella 6.4: Elenco delle scelte per l'agente di trasporto della posta (MTA) tra i pacchetti dell'archivio Debian

6.3.1 La configurazione di exim4

Per la posta di Internet attraverso uno smarthost, riconfigurare i pacchetti `exim4-*` nel modo seguente.

```
$ sudo /etc/init.d/exim4 stop
$ sudo dpkg-reconfigure exim4-config
```

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".

Selezionare "<No>" per "Omettere il mail name locale dai messaggi in uscita?". (Usare invece `/etc/email-addresses` come in Sezione 6.3.3.)

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/".

Selezionare "<Sì>" per "Dividere la configurazione in molti piccoli file?".

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:nomeutente@hostname.dom:password
```

Avviare `exim4` con il comando seguente.

```
$ sudo /etc/init.d/exim4 start
```

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.)
- Eseguire `invoke-rc.d exim4 reload`.

Leggere la guida ufficiale in `"/usr/share/doc/exim4-base/README.Debian.gz"` e `update-exim4.conf(8)`.



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.



avvertimento

Usare password in testo semplice senza cifratura non è sicuro, nemmeno se il proprio ISP lo permette.

Suggerimento

Benché sia raccomandato l'uso di [SMTP](#) con [STARTTLS](#) sulla porta 587, alcuni ISP usano ancora il deprecato [SMTPS](#) (SSL sulla porta 465). `Exim4` dopo la versione 4.77 supporta questo protocollo `SMTPS` deprecato sia per il client sia per il server.

Suggerimento

Se si sta cercando un MTA leggero che rispetti `"/etc/aliases"` per il proprio PC portatile, si dovrebbe considerare la configurazione di `exim4(8)` con `"QUEUERUNNER='queueonly'"`, `"QUEUERUNNER='nodaemon'"` ecc. in `"/etc/default/exim4"`.

6.3.2 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 /etc/init.d/postfix stop
$ 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      nomeutente:password
$ sudo postmap hash:/etc/postfix/sasl_passwd
```

Avviare postfix con il comando seguente.

```
$ sudo /etc/init.d/postfix start
```

In questo caso l'uso di "[" e "]" 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`.

6.3.3 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 programma 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"`.

6.3.4 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
<code>N/D</code>	<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/*"`.

6.4 Programma di posta (MUA, Mail User Agent)

Se ci si iscrive ad una mailing list relativa a Debian, può essere una buona idea usare programmi di posta come `mutt` e `mew` che sono lo standard de facto per i partecipanti e che si comportano come previsto.

pacchetto	popcon	dimensione	tipo
evolution	V:40, I:280	440	programma per X con interfaccia utente grafica (GNOME3, suite groupware)
icedove	V:34, I:109	292 243	programma per X con interfaccia utente grafica (GNOME2, Mozilla Thunderbird senza marchi)
kmail	V:44, I:96	17211	programma per X con interfaccia utente grafica (KDE)
mutt	V:192, I:613	5938	programma basato su terminale a caratteri probabilmente usato con vim
mew	V:0, I:0	2328	programma basato su terminale a caratteri in (x)emacs

Tabella 6.8: Elenco di programmi di posta (MUA)

6.4.1 Programma di posta base —Mutt

Per usare `mutt` come programma di posta (MUA) in combinazione con `vim`, personalizzare `~/ .muttrc` nel modo seguente.

```
#
# File di configurazione utente per sovrascrivere le impostazioni in /etc/Muttrc
#
# falsifica gli indirizzi di posta sorgenti
set use_from
set hostname=esempio.dom
set from="Nome Cognome <nomeutente@esempio.dom>"
set signature="~/ .signature"

# vim: "gq" per riformattare le citazioni
set editor="vim -c 'set tw=72 et ft=mail'"

# "mutt" apre la posta in entrata, mentre "mutt -y" elenca le caselle di posta
set mbox_type=Maildir          # usa il formato Maildir di qmail per creare mbox
set mbox=~/.Mail              # mette tutte le caselle di posta in $HOME/Mail/
set spoolfile=+Inbox           # posta consegnata in $HOME/Mail/Inbox
set record=+Outbox             # salva la posta fcc in $HOME/Mail/Outbox
set postponed=+Postponed      # mette la posta posposta in $HOME/Mail/postponed
set move=no                   # non spostare gli elementi nella posta in entrata in mbox
set quit=ask-yes              # non uscire con il semplice uso di "q"
set delete=yes                # all'uscita cancella sempre senza chiedere
set fcc_clear                  # salva fcc come non cifrate

# Caselle di posta in Maildir (aggiornamento automatico)
mailboxes 'cd ~/.Mail; /bin/ls -l|sed -e 's/^/+/' | tr "\n" " "'
unmailboxes Maillog *.ev-summary

## Predefinito
#set index_format="%4C %Z %{%b %d} %-15.15L (%4l) %s"
## Indice con thread con mittenti (collassato)
set index_format="%4C %Z %{%b %d} %-15.15n %?M?(<#03M>)&(%4l)? %s"

## Predefinito
#set folder_format="%2C %t %N %F %2l %-8.8u %-8.8g %8s %d %f"
## solo i nomi delle cartelle
set folder_format="%2C %t %N %f"
```

Per visualizzare la posta HTML e gli allegati MS Word all'interno del messaggio, aggiungere a `/etc/mailcap` o a `~/ .mailcap` le righe seguenti.

```
text/html; lynx -force_html %s; needsterminal;
application/msword; /usr/bin/antiword '%s'; copiousoutput; description="Microsoft Word Text ↵
"; nametemplate=%s.doc
```

Suggerimento

Mutt può essere usato come client [IMAP](#) e convertitore di formato per le caselle di posta. Si possono marcare messaggi con "t", "T", ecc. Questi messaggi marcati possono essere copiati con ";"C" tra caselle di posta diverse e cancellati in un'unica volta con ";"d".

6.5 L'utilità per il recupero della posta da remoto e l'inoltro

Invece di eseguire un programma di posta per accedere alla posta remota ed elaborarli a mano, si può volere automatizzare questo processo per avere tutti i messaggi di posta consegnati all'host locale. L'utilità di recupero e inoltro della posta remota è lo strumento adatto allo scopo.

Anche se fetchmail(1) è stato lo standard de facto in GNU/Linux per il recupero della posta da remoto, l'autore preferisce getmail(1). Se si desidera rifiutare la posta prima di scaricarla per salvare banda, potrebbero essere utili [mailfilter](#) o [mpop](#). Qualsiasi strumento di recupero della posta si usi, è una buona idea configurare il sistema in modo che la posta recuperata venga consegnata ad un MDA, come [maildrop](#), attraverso una pipe.

pacchetto	popcon	dimensione	descrizione
fetchmail	V:7, I:20	2343	strumento di recupero della posta (POP3, APOP, IMAP) (vecchio)
getmail4	V:3, I:11	31 637	strumento di recupero della posta (POP3, IMAP4 e SDPS) (semplice, sicuro e affidabile)
mailfilter	V:0, I:0	274	strumento di recupero della posta (POP3) con capacità di filtraggio con espressioni regolari
mpop	V:0, I:0	222	strumento di recupero della posta (POP3) e MDA con capacità di filtraggio con espressioni regolari

Tabella 6.9: Elenco di utilità per il recupero della posta da remoto e l'inoltro

6.5.1 Configurazione di getmail

La configurazione di getmail(1) è descritta nella [documentazione di getmail](#). Quelle seguenti sono le mie impostazioni per accedere ad account POP3 multipli come utente.

Creare il file `"/usr/local/bin/getmails"` nel modo seguente.

```
#!/bin/sh
set -e
if [ -f $HOME/.getmail/running ]; then
    echo "getmail is already running ... (if not, remove $HOME/.getmail/running)" >&2
    pgrep -l "getmai[l]"
    exit 1
else
    echo "getmail has not been running ... " >&2
fi
if [ -f $HOME/.getmail/stop ]; then
    echo "do not run getmail ... (if not, remove $HOME/.getmail/stop)" >&2
    exit
fi
if [ "$x$1" = "x-l" ]; then
    exit
fi
rcfiles="/usr/bin/getmail"
for file in $HOME/.getmail/config/* ; do
    rcfiles="$rcfiles --rcfile $file"
done
date -u > $HOME/.getmail/running
```



```
eval "$srcfiles $@"  
rm $HOME/.getmail/running
```

Configurarlo nel modo seguente.

```
$ sudo chmod 755 /usr/local/bin/getmails  
$ mkdir -m 0700 $HOME/.getmail  
$ mkdir -m 0700 $HOME/.getmail/config  
$ mkdir -m 0700 $HOME/.getmail/log
```

Creare i file "\$HOME/.getmail/config/nome_pop3", uno per ciascun account POP3, nel modo seguente.

```
[retriever]  
type = SimplePOP3SSLRetriever  
server = pop.example.com  
username = nome_pop3@example.com  
password = <propria-password>  
  
[destination]  
type = MDA_external  
path = /usr/bin/maildrop  
unixfrom = True  
  
[options]  
verbose = 0  
delete = True  
delivered_to = False  
message_log = ~/.getmail/log/nome_pop3.log
```

Configurarlo nel modo seguente.

```
$ chmod 0600 $HOME/.getmail/config/*
```

Pianificare l'esecuzione di "/usr/local/bin/getmails" ogni 15 minuti usando cron(8), eseguendo "sudo crontab -e -u <nome_utente>" e aggiungendo la riga seguente alle voci di cron dell'utente.

```
5,20,35,50 * * * * /usr/local/bin/getmails --quiet
```

Suggerimento

Eventuali problemi nell'accesso POP3 potrebbero non derivare da getmail. Alcuni servizi POP3 gratuiti popolari possono violare il protocollo POP3 e i loro filtri anti-SPAM possono non essere perfetti. Per esempio, possono cancellare i messaggi subito dopo aver ricevuto il comando RETR, prima di ricevere il comando DELE, e possono mettere messaggi in quarantena nella casella dello Spam. Si dovrebbero minimizzare i danni configurandoli in modo da archiviare i messaggi a cui si ha avuto accesso, senza cancellarli. Vedere anche [Alcuni messaggi di posta non sono stati scaricati](#).

6.5.2 configurazione di fetchmail

La configurazione di fetchmail(1) è determinata da "/etc/default/fetchmail", "/etc/fetchmailrc" e "\$HOME/.fetchmailrc". Vedere il suo esempio di configurazione in "/usr/share/doc/fetchmail/examples/fetchmailrc.example".

6.6 Agente di consegna della posta (MDA, Mail Delivery Agent) con filtri

La maggior parte dei programmi MTA, come postfix e exim4, funziona anche da MDA (Mail Delivery Agent, agente di consegna della posta). Ci sono MDA specializzati con capacità di filtraggio.

Anche se procmail(1) è stato lo standard de facto in GNU/Linux per ciò riguarda gli MDA con filtri, l'autore preferisce maildrop(1). Qualsiasi strumento di filtraggio si usi, è una buona idea configurare il sistema per inviare la posta filtrata ad una [Maildir in stile gmail](#).

pacchetto	popcon	dimensione	descrizione
procmail	V:70, I:618	300	MDA con filtri (vecchio)
mailagent	V:0, I:11	1283	MDA con filtri Perl
maildrop	V:1, I:3	1092	MDA con linguaggio strutturato per i filtri

Tabella 6.10: Elenco di MDA con filtri

6.6.1 configurazione di maildrop

La configurazione di maildrop(1) è descritta nella [documentazione di maildropfilter](#). Quello che segue è un esempio di configurazione per "\$HOME/.mailfilter".

```
# Configurazione locale
MAILROOT="$HOME/Mail"
# impostare questo al contenuto di /etc/mailname
MAILHOST="esempio.dom"
logfile $HOME/.maildroplog

# le regole sono fatte per sovrascrivere il valore precedente con quello successivo.

# posta di mailing list ?
if (    /^Precedence:.*list/:h || /^Precedence:.*bulk/:h )
{
    # regole per la posta di mailing list
    # casella di posta predefinita per i messaggi di mailing list
    MAILBOX="Inbox-list"
    # casella di posta predefinita per i messaggi di debian.org
    if ( /^(Sender|Resent-From|Resent-Sender): .*debian.org/:h )
    {
        MAILBOX="service.debian.org"
    }
    # casella di posta predefinita per i messaggi di bugs.debian.org (BTS)
    if ( /^(Sender|Resent-From|Resent-sender): .*@bugs.debian.org/:h )
    {
        MAILBOX="bugs.debian.org"
    }
    # casella di posta per ogni mailing list correttamente mantenuta con "List-Id: pippo" o ←
    "List-Id: ...<pippo.pluto>"
    if ( /^List-Id: ([^<]*<)?([^>]*>)?/:h )
    {
        MAILBOX="$MATCH2"
    }
}
else
{
    # regole per messaggi non di mailing list
    # casella di posta in entrata predefinita
    MAILBOX="Inbox-unusual"
    # messaggi di posta locali
    if ( /Envelope-to: .*@$MAILHOST/:h )
    {
        MAILBOX="Inbox-local"
    }
    # messaggi di posta html (al 99% posta indesiderata)
    if ( /DOCTYPE html/:b ||\
```

```

    /^Content-Type: text\/html/ )
{
    MAILBOX="Inbox-html"
}
# regola per la lista nera per la posta indesiderata
if ( /^X-Advertisement/:h ||\
    /^Subject:.*BUSINESS PROPOSAL/:h ||\
    /^Subject:.*URGENT.*ASISSTANCE/:h ||\
    /^Subject:.*I NEED YOUR ASSISTANCE/:h )
{
    MAILBOX="Inbox-trash"
}
# regola per la lista bianca per la posta normale
if ( /^From:.*@debian.org/:h ||\
    /^(Sender|Resent-From|Resent-Sender):.*debian.org/:h ||\
    /^Subject:.*(debian|bug|PATCH)/:h )
{
    MAILBOX="Inbox"
}
# regola per la lista bianca dei messaggi relativi al BTS
if ( /^Subject:.*Bug#.*/:h ||\
    /^(To|Cc):.*@bugs.debian.org/:h )
{
    MAILBOX="bugs.debian.org"
}
# regola per la lista bianca per messaggi di cron relativi a getmail
if ( /^Subject: Cron .*getmails/:h )
{
    MAILBOX="Inbox-getmails"
}
}

# controlla esistenza di $MAILBOX
'test -d $MAILROOT/$MAILBOX'
if ( $RETURNCODE == 1 )
{
    # crea una casella di posta maildir per $MAILBOX
    'maildirmake $MAILROOT/$MAILBOX'
}
# consegna nella maildir $MAILBOX
to "$MAILROOT/$MAILBOX/"
exit

```



avvertimento

A differenza di procmail, maildrop non crea automaticamente le directory maildir mancanti. Devono essere create prima manualmente usando maildirmake(1) come nell'esempio di file "\$HOME/.mailfilter" precedente.

6.6.2 configurazione di procmail

Quello che segue è un file di configurazione simile "\$HOME/.procmailrc" per procmail(1).

```

MAILDIR=$HOME/Maildir
DEFAULT=$MAILDIR/Inbox/
LOGFILE=$MAILDIR/Maillog
# posta palesemente con brutto aspetto: gettarla in X-trash e uscire
:0

```

```

* 1^0 ^X-Advertisement
* 1^0 ^Subject:.*BUSINESS PROPOSAL
* 1^0 ^Subject:.*URGENT.*ASISSTANCE
* 1^0 ^Subject: *I NEED YOUR ASSISTANCE
X-trash/

# Consegna dei messaggi di mailinglist
:0
* 1^0 ^Precedence:.*list
* 1^0 ^Precedence:.*bulk
* 1^0 ^List-
* 1^0 ^X-Distribution:.*bulk
{
:0
* 1^0 ^Return-path:.*debian-devel-admin@debian.or.jp
jp-debian-devel/

:0
* ^Resent-Sender.*debian-user-request@lists.debian.org
debian-user/

:0
* ^Resent-Sender.*debian-devel-request@lists.debian.org
debian-devel/

:0
* ^Resent-Sender.*debian-announce-request@lists.debian.org
debian-announce

:0
mailing-list/
}

:0
Inbox/

```

6.6.3 Riconsegna del contenuto di mbox

Se la propria directory home diventa piena e la consegna con procmail(1) fallisce, sarà necessario consegnare manualmente la posta da `/var/mail/<nomeutente>` alle proprie caselle di posta distinte nella directory home. Dopo aver fatto spazio nella directory home, eseguire quanto segue.

```

# /etc/init.d/${MAILDAEMON} stop
# formail -s procmail </var/mail/<nomeutente>
# /etc/init.d/${MAILDAEMON} start

```

6.7 Server POP3/IMAP4

Se si deve eseguire un server privato in una LAN, si può prendere in considerazione l'esecuzione di un server [POP3](#) / [IMAP4](#) per la consegna della posta ai client nella LAN.

6.8 Server ed utilità per la stampa

Nei vecchi sistemi *nix, lo standard era il [demone di stampa](#) BSD. Dato che il formato standard per l'output di stampa del software libero nei sistemi in stile Unix è PostScript, veniva usato un sistema di filtri insieme a [Ghostscript](#) per permettere la stampa su stampanti non-PostScript.

pacchetto	popcon	dimensione	tipo	descrizione
courier-pop	V:3, I:5	239	POP3	Server di posta Courier - server POP3 (solo formato maildir)
cyrus-pop3d	V:0, I:0	147	POP3	sistema di posta Cyrus (supporto POP3)
courier-imap	V:6, I:7	544	IMAP	Server di posta Courier - server IMAP (solo formato maildir)
cyrus-imapd	V:0, I:1	332	IMAP	sistema di posta Cyrus (supporto IMAP)

Tabella 6.11: Elenco di server POP3/IMAP4

Di recente il nuovo standard de facto è [CUPS \(Common UNIX Printing System\)](#). CUPS usa il protocollo [IPP \(Internet Printing Protocol\)](#). IPP è ora supportato da altri sistemi operativi come Windows XP e Mac OS X ed è diventato il nuovo standard de facto multiplatforma per la stampa da remoto con capacità di comunicazione bidirezionale.

Il formato standard per i dati stampabili per le applicazioni nel sistema Debian è il formato [PostScript \(PS\)](#) che è un linguaggio di descrizione della pagina. I dati in formato PS vengono passati all'interprete PosPostScript Ghostscript per produrre i dati stampabili specifici per la stampante. Vedere Sezione [11.4.1](#).

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:7, I:8	324	printer (515)	<code>lpr/lpd</code> BSD (Line printer daemon , demone di stampa in linea)
lprng	V:1, I:1	3852	" "	" " (Migliorato)
cups	V:250, I:441	764	IPP (631)	server CUPS di stampa Internet
cups-client	V:67, I:496	473	" "	comandi System V per la stampa con CUPS: <code>lp(1)</code> , <code>lpstat(1)</code> , <code>lpoptions(1)</code> , <code>cancel(1)</code> , <code>lpmove(8)</code> , <code>lpinfo(8)</code> , <code>lpadmin(8)</code> , ...
cups-bsd	V:37, I:424	115	" "	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:107, I:423	881	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/>".

6.9 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`.



Attenzione

Se il proprio SSH è accessibile da Internet, vedere Sezione [4.7.3](#).

pacchetto	popcon	dimensione	strumento	descrizione
openssh-client	V:822, I:995	4147	ssh(1)	client SSH
openssh-server	V:684, I:811	870	sshd(8)	server SSH
ssh-askpass-fullscreen	screen V:0, I:1	42	ssh-askpass-fullscreen(1)	chiede all'utente una passphrase per ssh-add (GNOME2)
ssh-askpass	V:3, I:42	101	ssh-askpass(1)	chiede all'utente una passphrase per ssh-add (X semplice)

Tabella 6.13: Elenco dei server e delle utilità per l'accesso remoto

Suggerimento

Per permettere al processo della shell remota di sopravvivere all'interruzione della connessione, usare il programma `screen(1)` (vedere Sezione [9.1](#)).

6.9.1 Nozioni di base su SSH**avvertimento**

Se si desidera eseguire il server OpenSSH, non deve esistere il file `"/etc/ssh/sshd_not_to_be_run"`.

SSH ha due protocolli di autenticazione

protocollo SSH	metodo SSH	descrizione
SSH-1	"RSAAuthentication"	autenticazione utente RSA basata su chiave identificativa
" "	"RhostsAuthentication"	autenticazione di host basata su <code>".rhosts"</code> (insicura, disabilitata)
" "	"RhostsRSAAuthentication"	autenticazione di host basata su <code>".rhosts"</code> in combinazione con chiave RSA per host(disabilitata)
" "	"ChallengeResponseAuthentication"	autenticazione RSA challenge-response
" "	"PasswordAuthentication"	autenticazione basata su password
SSH-2	"PubkeyAuthentication"	autenticazione utente basata su chiave pubblica
" "	"HostbasedAuthentication"	autenticazione di host basata su <code>"~/ .rhosts"</code> o <code>"/etc/hosts.equiv"</code> in combinazione con autenticazione dell'host client con chiave pubblica (disabilitata)
" "	"ChallengeResponseAuthentication"	autenticazione challenge-response
" "	"PasswordAuthentication"	autenticazione basata su password

Tabella 6.14: Elenco dei protocolli e dei metodi di autenticazione di SSH

**Attenzione**

Se si usa un sistema non Debian si faccia attenzione a queste differenze.

file di configurazione	descrizione del file di configurazione
/etc/ssh/ssh_config	impostazioni predefinite per il client SSH, vedere ssh_config(5)
/etc/ssh/sshd_config	impostazioni predefinite per il server SSH, vedere ssh_config(5)
~/.ssh/authorized_keys	chiavi SSH pubbliche predefinite che i client usano per connettersi a questo account su questo server SSH
~/.ssh/identity	chiave RSA SSH-1 segreta dell'utente
~/.ssh/id_rsa	chiave RSA SSH-2 segreta dell'utente
~/.ssh/id_dsa	chiave DSA SSH-2 segreta dell'utente

Tabella 6.15: Elenco dei file di configurazione per SSH

Per i dettagli vedere `"/usr/share/doc/ssh/README.Debian.gz"`, ssh(1), sshd(8), ssh-agent(1) e ssh-keygen(1).

Di seguito è riportato l'elenco dei file di configurazione più importanti.

Suggerimento

Per l'uso delle chiavi SSH pubbliche e segrete, vedere ssh-keygen(1), ssh-add(1) e ssh-agent(1).

Suggerimento

Assicurarsi di verificare le impostazioni testando la connessione. In caso di problemi, usare `"ssh -v"`.

Suggerimento

Si può cambiare successivamente la passphrase per cifrare le chiavi SSH segrete locali usando `"ssh-keygen -p"`.

Suggerimento

Si possono aggiungere opzioni alle voci in `~/.ssh/authorized_keys` per limitare gli host e per eseguire comandi specifici. Per i dettagli vedere sshd(8).

I comandi seguenti avviano una connessione ssh(1) da un client.

comando	descrizione
ssh nomeutente@nomehost.dominio.ext	connette nella modalità predefinita
ssh -v nomeutente@nomehost.dominio.ext	connette nella modalità predefinita con messaggi di debug
ssh -1 nomeutente@nomehost.dominio.ext	forza la connessione con SSH versione 1
ssh -1 -o RSAAuthentication=no -l nomeutente nomehost.dominio.ext	forza l'uso di password con SSH versione 1
ssh -o PreferredAuthentications=password -l nomeutente nomehost.dominio.ext	forza l'uso di password con SSH versione 2

Tabella 6.16: Elenco di esempi di avvio di client SSH

Se si usa lo stesso nome utente sull'host locale e su quello remoto, si può evitare di digitare la parte `"nomeutente@"`. Anche se si usa un nome utente diverso nell'host locale rispetto a quello remoto, si può evitare di digitare quella parte usando il file

"~/.ssh/config". Per il [servizio Debian Alioth](#) con nome account "pippo-guest", impostare "~/.ssh/config" in modo che contenga quanto segue.

```
Host alioth.debian.org svn.debian.org git.debian.org
  User pippo-guest
```

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-]).

6.9.2 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:server-remoto:25 4110:server-remoto:110 nomeutente@server-remoto
```

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".

6.9.3 Connettersi senza password remote

Si può evitare di doversi ricordare le password per sistemi remoti usando "RSAAuthentication" (protocollo SSH-1) o "PubkeyAuthentication" (protocollo SSH-2).

Sul sistema remoto impostare nel file "/etc/ssh/sshd_config" le rispettive voci "RSAAuthentication yes" o "PubkeyAuthentication yes".

Generare localmente le chiavi di autenticazione ed installare la chiave pubblica sul sistema remoto con i comandi seguenti.

- "RSAAuthentication": chiave RSA per SSH-1 (deprecata perché superata)

```
$ ssh-keygen
$ cat .ssh/identity.pub | ssh utente1@remote "cat - >>.ssh/authorized_keys"
```

- "PubkeyAuthentication": chiave RSA per SSH-2

```
$ ssh-keygen -t rsa
$ cat .ssh/id_rsa.pub | ssh utente1@remote "cat - >>.ssh/authorized_keys"
```

- "PubkeyAuthentication": chiave DSA per SSH-2 (deprecata perché lenta)

```
$ ssh-keygen -t dsa
$ cat .ssh/id_dsa.pub | ssh utente1@remote "cat - >>.ssh/authorized_keys"
```

Suggerimento

L'uso di una chiave DSA per SSH-2 è deprecato perché la chiave è più piccola e più lenta. Non esistono più motivi per aggirare i brevetti RSA usando DSA, dato che essi sono scaduti. DSA sta per [Algoritmo di firma digitale \(Digital Signature Algorithm\)](#) ed è lento. Vedere anche [DSA-1571-1](#).

Nota

Affinché "HostbasedAuthentication" funzioni con SSH-2, si deve impostare la voce "HostbasedAuthentication" a "yes" in entrambi i file "/etc/ssh/sshd_config" nell'host server e i "/etc/ssh/ssh_config" o "~/.ssh/config" nell'host client.

6.9.4 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 6.17: Elenco di client SSH per altre piattaforme

6.9.5 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/<nomeutente>/.ssh/id_rsa:
Identity added: /home/<nomeutente>/.ssh/id_rsa (/home/<nomeutente>/.ssh/id_rsa)
```

Per il comando successivo non sarà più necessaria da questo momento la password remota.

```
$ scp pippo <nomeutente>@host.remoto:pippo
```

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)`.

6.9.6 Spegner 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 [9.3.13](#)) nel modo seguente.

```
# echo "shutdown -h now" | at now
```

Eseguire `shutdown -h now` in una sessione `screen(1)` (vedere Sezione [9.1](#)) è un altro modo di ottenere lo stesso risultato.

6.9.7 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.

pacchetto	popcon	dimensione	protocollo	descrizione
telnetd	V:0, I:3	103	TELNET	server TELNET
telnetd-ssl	V:0, I:0	154	” ”	” ” (supporto SSL)
nfs-kernel-server	V:33, I:92	322	NFS	condivisione di file Unix
samba	V:112, I:189	10994	SMB	condivisione di file e stampanti Windows
netatalk	V:2, I:5	3923	ATP	condivisione di file e stampanti Apple/Mac (AppleTalk)
proftpd-basic	V:34, I:42	4406	FTP	scaricamento generico di file
apache2	V:232, I:351	583	HTTP	server web generico
squid	V:13, I:16	7093	” ”	server proxy web generico
squid3	V:12, I:18	162	” ”	” ”
bind9	V:55, I:80	1891	DNS	indirizzo IP per altri host
isc-dhcp-server	V:24, I:62	1389	DHCP	indirizzo IP per il client stesso

Tabella 6.18: Elenco di altri server applicativi di rete

6.10 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.

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 molto più efficiente per risparmiare banda rispetto all'uso di server mirror locali con il completo contenuto dell'archivio Debian.

6.11 Altri client di rete

Ecco un elenco di altri client applicativi di rete.

6.12 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.nomeisp.net pop3
```

pacchetto	popcon	dimensione	protocollo	descrizione
netcat	I:34	16	TCP/IP	coltellino svizzero TCP/IP
openssl	V:794, I:990	1303	SSL	eseguibile SSL (Secure Socket Layer) e strumenti crittografici relativi
stunnel4	V:4, I:18	461	” ”	wrapper SSL universale
telnet	V:69, I:908	157	TELNET	client TELNET
telnet-ssl	V:1, I:6	208	” ”	” ” (supporto SSL)
nfs-common	V:328, I:600	722	NFS	condivisione di file Unix
smbclient	V:20, I:213	1706	SMB	client per condivisione di file e stampanti MS Windows
cifs-utils	V:36, I:122	230	” ”	comandi mount e umount per file MS Windows remoti
ftp	V:32, I:602	132	FTP	client FTP
lftp	V:6, I:41	2198	” ”	” ”
ncftp	V:4, I:29	1506	” ”	client FTP a tutto schermo
wget	V:351, I:998	3104	HTTP e FTP	scaricatore per web
curl	V:117, I:452	376	” ”	” ”
axel	V:0, I:6	186	” ”	scaricatore accelerato
aria2	V:2, I:17	4557	” ”	scaricatore accelerato con gestione di BitTorrent e Metalink
bind9-host	V:452, I:949	171	DNS	host(1) da bind9, ”Priorità: standard”
dnsutils	V:74, I:721	440	” ”	dig(1) da bind, ”Priorità: standard”
isc-dhcp-client	V:299, I:968	650	DHCP	ottiene indirizzo IP
ldap-utils	V:18, I:78	680	LDAP	ottiene dati da server LDAP

Tabella 6.19: Elenco di altri client applicativi di rete

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.

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 6.20: Elenco di RFC popolari

L'uso delle porte è descritto in `"/etc/services"`.

Capitolo 7

Il sistema X Window



avvertimento

Questo capitolo sta diventando non aggiornato dato che è basato su Debian 7.0 (wheezy), rilasciata nel 2013.

Il [sistema X Window](#) in Debian è basato sui sorgenti di [X.org](#).

7.1 Pacchetti fondamentali

Per facilitare l'installazione sono forniti alcuni (meta)pacchetti

(meta)pacchetto	popcon	dimensione	descrizione
xorg	I:503	52	librerie X, un server X, un insieme di tipi di carattere e un gruppo di client e utilità X di base (metapacchetto)
xserver-xorg	V:109, I:544	238	insieme completo del server X e della sua configurazione
xbase-clients	I:53	46	vari client X assortiti (metapacchetto)
x11-common	V:390, I:801	305	infrastruttura del filesystem per il sistema X Window
xorg-docs	I:7	2032	documentazione varia per la suite software X.Org
menu	V:111, I:285	1435	genera il menu Debian per tutte le applicazioni con supporto per il menu
gksu	V:41, I:259	289	frontend Gtk+ a su(1) o sudo(8)
menu-xdg	V:34, I:139	76	converte la struttura del menu Debian nella struttura del menu xdg di freedesktop.org
xdg-utils	V:260, I:564	325	utilità per integrare l'ambiente desktop fornito da freedesktop.org
task-gnome-desktop	I:207	6	ambiente desktop GNOME standard (metapacchetto)
task-kde-desktop	I:68	6	ambiente desktop KDE di base (metapacchetto)
task-xfce-desktop	I:102	6	ambiente desktop leggero Xfce (metapacchetto)
task-lxde-desktop	I:44	6	ambiente desktop leggero LXDE (metapacchetto)
fluxbox	V:2, I:12	3860	Fluxbox : pacchetto per un gestore di finestre per X altamente configurabile e con basso uso di risorse

Tabella 7.1: Elenco di (meta)pacchetti fondamentali per X Window

Per le nozioni base su X fare riferimento a X(7) e il [XWindow-User-HOWTO dell'LDP](#).

7.2 Impostare l'ambiente desktop

Un [ambiente desktop](#) è solitamente una combinazione di un [gestore di finestre per X](#), un gestore di file e una suite di utilità compatibili.

Si può configurare un [ambiente desktop](#) completo come [GNOME](#), [KDE](#), [Xfce](#) o [LXDE](#) da aptitude nel menu dei task.

Suggerimento

Il menu dei task potrebbe non essere in sincrono con le più recenti transizioni di pacchetti negli ambienti Debian `unstable/testing`. In questo caso è necessario deselezionare alcuni (meta)pacchetti elencati nel menu dei task di aptitude(8) per evitare conflitti tra pacchetti. Quando si deselezionano (meta)pacchetti, è necessario selezionare manualmente alcuni pacchetti che forniscono le loro dipendenze per evitare che vengano automaticamente rimossi.

In alternativa è possibile impostare manualmente un ambiente semplice con solamente un [gestore di finestre per X](#) come [Fluxbox](#).

Vedere [Gestori di finestre per X](#) per una guida ai gestori di finestre per X e agli ambienti desktop.

7.2.1 Menu Debian

Il [sistema del menu Debian](#) fornisce un'interfaccia generale ai programmi sia testuali sia orientati ad X, tramite `update-menus(1)` nel pacchetto `menu`. Ciascun pacchetto installa i propri dati per il menu nella directory `"/usr/share/menu/"`. Vedere `"/usr/share/menu/README"`.

7.2.2 Menu Freedesktop.org

Ogni pacchetto conforme al sistema di menu xdg di Freedesktop.org installa i propri dati di menu forniti da `"*.desktop"` in `"/usr/share/applications/"`. Gli ambienti desktop moderni che sono conformi agli standard Freedesktop.org usano questi dati per generare i propri menu usando il pacchetto `xdg-utils`. Vedere `"/usr/share/doc/xdg-utils/README"`.

7.2.3 Menu Debian dal menu di Freedesktop.org

Per poter accedere al tradizionale menu Debian da ambienti con gestori delle finestre conformi con il [menu di Freedesktop.org](#), come GNOME e KDE, è necessario installare il pacchetto `menu-xdg`.

7.3 La relazione client/server

Il sistema X Window è attivato come combinazione di programmi server e client. Il significato delle parole **server** e **client** in relazione alle parole **locale** e **remoto** richiede un po' di attenzione.

I server X moderni hanno l'[estensione MIT Shared Memory](#) e comunicano con i loro client X locali usando la memoria condivisa locale. Ciò scavalca il canale di comunicazione tra i processi Xlib trasparente alla rete e fa guadagnare in prestazioni per grandi immagini.

7.4 Il server X

Per informazioni sul server X vedere `xorg(1)`.

tipo	descrizione
server X	un programma eseguito su un host locale connesso al display dell'utente e ai device di input
client X	un programma in esecuzione su un host remoto che elabora dati e comunica con il server X
application server	un programma in esecuzione su un host remoto che elabora dati e comunica con le applicazioni client.
client applicativo	un programma eseguito su un host locale connesso al display dell'utente e ai device di input

Tabella 7.2: Elenco della terminologia per server/client

7.4.1 La (ri)configurazione del server X

Quanto segue (ri)configura un server X.

```
# dpkg-reconfigure --priority=low x11-common
```

Nota

I kernel Linux recenti hanno un buon supporto per i dispositivi grafici e di input con [DRM](#), [KMS](#) e [udev](#). Il server X è stato riscritto per usarli. Perciò `/etc/X11/xorg.conf` è solitamente non presente nel sistema. Questi parametri vengono configurati dal kernel. Vedere `fb/modedb.txt` nella documentazione del kernel.

Per monitor CRT grandi con alte risoluzioni, è una buona idea impostare la frequenza di aggiornamento al valore più alto gestito dal monitor (85 Hz è un valore ottimo, 75 Hz è OK) per ridurre lo sfarfallio. Per i monitor LCD frequenze di aggiornamento standard più basse (60Hz) vanno solitamente bene a causa della risposta più lenta.

Nota

Fare attenzione a non usare una frequenza di aggiornamento troppo alta che potrebbe causare danni hardware critici al monitor.

7.4.2 I metodi di connessione al server X

Ci sono diversi modi per far sì che il "server X" (lato **display**) accetti connessioni da un "client X" (lato **applicazione**).



avvertimento

Non usare connessioni remote [TCP/IP](#) su reti **non sicure** per la connessione X a meno che non esistano ragioni molto buone per farlo, come l'uso di cifratura. Una connessione remota via socket TCP/IP senza cifratura è esposta ad **attacchi di spionaggio** ed è disabilitata in modo predefinito nel sistema Debian. Usare `ssh -X`.



avvertimento

Non usare nemmeno [connessioni XDMCP](#) su reti **non sicure**. Inviano dati attraverso [UDP/ IP](#) senza cifratura e sono esposte ad **attacchi di spionaggio**.

Suggerimento

LTSP sta per [Linux Terminal Server Project](#) (progetto per server terminale Linux).

pacchetto	popcon	dimensione	controllato	cifratura	metodo	uso collegato
xbase-clients	I:53	46	non controllato	no	comando xhost	deprecato
xbase-clients	I:53	46	controllato	no	comando xauth	connessione locale attraverso una pipe
openssh-client	V:822, I:995	4147	controllato	sì	comando ssh -X	connessione di rete remota
gdm3	V:187, I:277	4697	controllato	no (XDMCP)	GNOME display manager	connessione locale attraverso una pipe
sddm	V:34, I:65	1661	controllato	no (XDMCP)	KDE display manager	connessione locale attraverso una pipe
xdm	V:4, I:9	665	controllato	no (XDMCP)	gestore dello schermo di X	connessione locale attraverso una pipe
wdm	V:184, I:605	2301	controllato	no (XDMCP)	WindowMaker display manager	connessione locale attraverso una pipe
lwm	V:0, I:1	413	controllato	sì	LTSP display manager	connessione SSH di rete remota (thin client)

Tabella 7.3: Elenco di metodi di connessione al server X

7.5 Avviare il sistema X Window

Il sistema X Window è solitamente avviato come una [sessione X](#), che è la combinazione di un server X e di client X connessi. Per un normale sistema desktop entrambe le componenti sono eseguite su una macchina.

La [sessione X](#) viene avviata in uno dei modi seguenti.

- Il comando `startx` avviato dalla riga di comando
- Uno dei programmi demone [display manager per X](#), *`dm` avviati al termine dello script di avvio nella directory `"/etc/rc?.d/"` (dove `"?"` corrisponde al runlevel)

Suggerimento

Lo script di avvio dei demoni display manager controlla il contenuto del file `"/etc/X11/default-display-manager"` prima dell'effettivo avvio del demone. ciò assicura che venga attivato un solo programma demone per [display manager di X](#).

Suggerimento

Vedere Sezione [8.4.5](#) per le variabili d'ambiente iniziali del display manager per X.

Essenzialmente tutti questi programmi eseguono lo script `"/etc/X11/Xsession"`; poi questo script effettua un'azione simile a `run-parts(8)` per eseguire tutti gli script nella directory `"/etc/X11/Xsession.d/"`. Ciò è fondamentalmente l'esecuzione, con il comando interno `exec`, del primo programma trovato cercando nell'ordine seguente.

1. Lo script specificato come argomento di `"/etc/X11/Xsession"` dal display manager di X, se definito.
2. Lo script `"~/Xsession"` o `"~/Xsession"`, se definito.

3. Il comando `"/usr/bin/x-session-manager"`, se definito.
4. Il comando `"/usr/bin/x-window-manager"`, se definito.
5. Il comando `"/usr/bin/x-terminal-emulator"`, se definito.

Questo processo è influenzato dal contenuto di `"/etc/X11/Xsession.options"`. I programmi effettivi a cui puntano questi comandi `"/usr/bin/x-"` sono determinati dal sistema di alternative di Debian e possono essere modificati con `"update-alternatives --config x-session-manager"`, ecc.

Vedere `Xsession(5)` per i dettagli.

7.5.1 Avviare la sessione X con gdm3

`gdm3(1)` permette di selezionare dal suo menu il tipo di sessione (o l'ambiente desktop: Sezione 7.2) e la lingua (o la localizzazione: Sezione 8.4) della sessione X. Archivia il valore selezionato predefinito in `"~/ .dmrc"` nel formato seguente.

```
[Desktop]
Session=default
Language=ja_JP.UTF-8
```

7.5.2 Personalizzare la sessione X (metodo classico)

In un sistema in cui `"/etc/X11/Xsession.options"` contiene una riga `"allow-user-xsession"` non preceduta da nessun carattere `"#"`, qualsiasi utente che configuri `"~/ .xsession"` o `"~/ .Xsession"` è in grado di personalizzare il comportamento di `"/etc/X11/Xsession"` sovrascrivendo completamente il codice del sistema. L'ultimo comando nel file `"~/ .xsession"` dovrebbe essere nella forma `"exec un-qualche-gestore-di-finestre/sessione"` per avviare il proprio gestore di finestre/sessione di X preferito.

Se viene usata questa funzionalità, la selezione del gestore di display (o di login) (DM), del gestore di sessione o del gestore di finestre (WM) da parte dell'utilità di sistema viene ignorata.

7.5.3 Personalizzare la sessione X (metodo nuovo)

Quelli che seguono sono metodi nuovi per personalizzare la sessione X senza sovrascrivere completamente le impostazioni di sistema come nell'esempio precedente.

- Il display manager `gdm3` può selezionare una specifica sessione ed impostarla come argomento di `"/etc/X11/Xsession"`.
 - I file `"/etc/profile"`, `"~/ .profile"`, `"/etc/xprofile"` e `"~/ .xprofile"` sono eseguiti come parte del processo di avvio di `gdm3`.
- Il file `"~/ .xsessionrc"` è eseguito come parte del processo di avvio. (Indipendente dal desktop.)
 - `"#allow-user-xsession"` in `"/etc/X11/Xsession.options"` non restringe l'esecuzione del file `"~/ .xsessionrc"`.
- Il file `"~/ .gnomerc"` è eseguito come parte del processo di avvio (solo desktop GNOME)

La selezione del gestore di display (o di login) (DM), del gestore di sessione o del gestore di finestre (WM) da parte dell'utilità di sistema viene rispettata.

Questi file di configurazione non dovrebbero contenere `"exec ..."` né `"exit"`.

7.5.4 Connettersi ad un client X remoto via SSH

L'uso di `"ssh -X"` permette una connessione sicura da un server X locale ad un application server remoto.

Impostare le voci `"X11Forwarding"` a `"yes"` nel file `"/etc/ssh/sshd_config"` dell'host remoto, se si vuole evitare l'opzione `"-X"` nella riga di comando.

Avviare il server X sull'host locale

Aprire un `xterm` sull'host locale.

Eseguire `ssh(1)` per stabilire una connessione con il sito remoto nel modo seguente.

```
nomelocale @ hostlocale $ ssh -q -X nomelocal@hostremoto.dominio
Password:
```

Eseguire un comando applicativo X, ad esempio `"gimp"`, sul sito remoto nel modo seguente.

```
nomelocal @ hostremoto $ gimp &
```

Questo metodo può mostrare l'output da un client X remoto come se fosse connesso localmente attraverso un socket UNIX locale.

7.5.5 Terminale X sicuro via Internet

Un terminale X sicuro via Internet, che mostri in remoto l'intero ambiente desktop X, può essere ottenuto facilmente usando pacchetti specializzati, come `ldm`. La macchina locale diventa un thin client sicuro per l'application server remoto connesso via SSH.

7.6 Tipi di carattere in X Window

[Fontconfig 2.0](#) è stato creato nel 2002 per fornire una libreria indipendente dalla distribuzione per configurare e personalizzare l'accesso ai tipi di carattere. Debian a partire da `squeeze` usa [Fontconfig 2.0](#) per la sua configurazione.

Il supporto dei tipi di carattere nel sistema X Window può essere riassunto nel modo seguente.

- Sistema datato di supporto dei tipi di carattere dal lato server X
 - Il sistema originale base di X per i caratteri fornisce la compatibilità all'indietro per le versioni più vecchie dei client X.
 - I tipi di carattere originali base di X11 sono installati nel server X.
- Sistema moderno di supporto dei tipi di carattere dal lato client X
 - Il sistema X moderno supporta tutti i tipi di carattere elencati in seguito (Sezione [7.6.1](#), Sezione [7.6.2](#) e Sezione [7.6.3](#)) con funzionalità avanzate come l'anti-aliasing.
 - [Xft 2.0](#) connette applicazioni X moderne come quelle di [GNOME](#), [KDE](#) e [LibreOffice](#) con la libreria [FreeType 2.0](#).
 - [FreeType 2.0](#) fornisce la libreria per la rasterizzazione dei tipi di carattere.
 - [Fontconfig](#) fornisce la risoluzione delle specifiche per i tipi di carattere per [Xft 2.0](#). Per la sua configurazione vedere `fonts.conf(5)`.
 - tutte le moderne applicazioni che usano [Xft 2.0](#) possono comunicare con i server X moderni usando l'estensione [X Rendering Extension](#).
 - La [X Rendering Extension](#) sposta l'accesso ai tipi di carattere e la generazione delle immagini per i glifi dal server X al client X.

È possibile controllare le informazioni sulla configurazione dei tipi di carattere nei modi seguenti.

- `"xset q"` per il percorso dei tipi di carattere base di X11

pacchetto	popcon	dimensione	descrizione
xfonts-utils	V:38, I:608	387	programmi di utilità del sistema X Window per i tipi di carattere
libxft2	V:122, I:730	122	Xft, una libreria che connette le applicazioni X con la libreria di rasterizzazione dei caratteri FreeType
libfreetype6	V:452, I:993	920	libreria di rasterizzazione dei tipi di carattere FreeType 2.0
fontconfig	V:391, I:862	563	Fontconfig , una libreria generica per la configurazione dei tipi di caratteri - binari di supporto
fontconfig-config	V:394, I:918	472	Fontconfig , una libreria generica per la configurazione dei tipi di caratteri - dati di configurazione

Tabella 7.4: Tabella dei pacchetti che supportano i sistemi dei tipi di carattere in X Window

- "fc-match" per le impostazioni predefinite per i tipi di carattere di fontconfig
- "fc-list" per l'elenco dei tipi di carattere di fontconfig disponibili

Suggerimento

"[The Penguin and Unicode](#)" è una buona panoramica sul sistema X Window moderno. Altra documentazione all'indirizzo <http://unifont.org/> dovrebbe fornire buone informazioni su tipi di carattere Unicode, software che gestisce Unicode, internazionalizzazione e problematiche correlate all'uso di Unicode su sistemi operativi [free/libre/liberi-open source \(FLOSS\)](#).

7.6.1 Tipi di carattere base

Ci sono due tipi principali di [caratteri per computer](#).

- I tipi di carattere bitmap (buoni per rasterizzazione a basse risoluzioni)
- Tipi di carattere outline/stroke (buoni per rasterizzazione ad alte risoluzioni)

Mentre il ridimensionamento dei caratteri bitmap causa immagini dentellate, il ridimensionamento dei caratteri outline/stroke produce un'immagine pulita.

I caratteri bitmap nel sistema Debian sono di solito forniti da [file di caratteri bitmap pcf per X11](#) compressi, la cui estensione è ".pcf.gz".

I caratteri outline nel sistema Debian sono forniti dai tipi seguenti.

- File di caratteri [PostScript](#) Type 1, la cui estensione è ".pfb" (file di caratteri binario) e ".afm" (file della metrica dei caratteri).
- File di caratteri [TrueType](#) (o [OpenType](#)), la cui estensione è normalmente ".ttf".

Suggerimento

[OpenType](#) è pensato per rendere superati sia [TrueType](#) sia [PostScript](#) Type 1.

Suggerimento

I tipi di carattere [DejaVu](#) sono basati sui caratteri [Bitstream Vera](#) di cui sono un sovrainsieme.

pacchetto con i caratteri	popcon	dimensione	carattere sans serif	carattere serif	carattere monospace	origine del carattere
PostScript	N/D	N/D	Helvetica	Times	Courier	Adobe
gsfonts	I:692	4439	Nimbus Sans L	Nimbus Roman No9 L	Nimbus Mono L	URW (dimensione Adobe-compatibile)
gsfonts-x11	I:123	96	Nimbus Sans L	Nimbus Roman No9 L	Nimbus Mono L	supporto per i caratteri per X con caratteri PostScript Type 1
t1-cyrillic	I:23	4884	Free Helvetian	Free Times	Free Courier	URW esteso (dimensione Adobe-compatibile)
lmodern	I:141	33270	LMSans*	LMRoman*	LMTypewriter*	caratteri PostScript e OpenType scalabili basati su Computer Modern (da TeX)

Tabella 7.5: Tabella dei tipi di carattere [PostScript](#) Type 1

pacchetto con i caratteri	popcon	dimensione	carattere sans serif	carattere serif	carattere monospace	origine del carattere
ttf-mscorefonts-installer	V:1, I:83	125	Arial	Times New Roman	Courier New	Microsoft (dimensione Adobe-compatibile) (Installa dati non liberi)
fonts-liberation	I:569	2093	Liberation Sans	Liberation Serif	Liberation Mono	Progetto Liberation Fonts (dimensione Microsoft-compatibile)
fonts-freefont-ttf	V:44, I:364	10746	FreeSans	FreeSerif	FreeMono	GNU freefont (dimensione Microsoft-compatibile)
fonts-dejavu	I:503	39	DejaVu Sans	DejaVu Serif	DejaVu Sans Mono	DejaVu , Bitstream Vera con copertura di Unicode
fonts-dejavu-core	V:306, I:756	2954	DejaVu Sans	DejaVu Serif	DejaVu Sans Mono	DejaVu , Bitstream Vera con copertura di Unicode (sans, sans-bold, serif, serif-bold, mono, mono-bold)
fonts-dejavu-extra	I:528	7217	N/D	N/D	N/D	DejaVu , Bitstream Vera with Unicode coverage (oblique, italic, bold-oblique, bold-italic, condensed)
ttf-unifont	I:23	16943	N/D	N/D	unifont	GNU Unifont , con tutti i codici di carattere stampabili in Unicode 5.1 BMP (Basic Multilingual Plane)

Tabella 7.6: Tabella dei tipi di carattere [TrueType](#)

7.6.2 Tipi di carattere aggiuntivi

aptitude(8) aiuta a trovare facilmente caratteri aggiuntivi.

- Il corto elenco di pacchetti in "Task" → "Localizzazione"
- L'elenco unico dei pacchetti filtrato per i dati dei tipi di carattere usando un'espressione regolare sui debtag: "~Gmade-of::data:fo"
- L'elenco unico dei pacchetti filtrato per i pacchetti con caratteri BDF (bitmap) usando un'espressione regolare sui nomi di pacchetto: "~nxfonTS-"
- L'elenco unico dei pacchetti filtrato per i pacchetti con caratteri TrueType (outline) usando un'espressione regolare sui nomi di pacchetto: "~nttf-|~nfonTS-"

Dato che i tipi di carattere **Liberi** sono a volte limitati, l'installazione o la condivisione di caratteri TrueType commerciali è un'opzione da considerare per un utente Debian. Per facilitare ciò sono stati creati alcuni pacchetti di comodo.

- `mathematica-fonts`
- `fonts-mscorefonts-installer`

Si avrà un'ottima selezione di caratteri TrueType al prezzo della contaminazione del proprio sistema **Libero** con tipi di carattere non-Liberi.

7.6.3 Tipi di carattere CJK

In questa sezione ci sono alcune nozioni fondamentali centrate sull'uso di tipi di carattere con [caratteri CJK](#).

tipologia del carattere	nome del tipo di carattere in giapponese	nome del tipo di carattere in cinese	nome del tipo di carattere in coreano
sans-serif	gothic, ゴシック	hei, gothic	dodum, gulim, gothic
serif	mincho, 明朝	song, ming	batang

Tabella 7.7: Tabella delle parole chiave usate nei nomi dei tipi di carattere CJK per indicarne il tipo

I tipi di carattere con nomi come "VL PGothic", dove la "P" sta per carattere proporzionale, sono la versione proporzionale dei corrispondenti caratteri a spaziatura fissa, come "VL Gothic".

Per esempio la tabella dei codici [Shift_JIS](#) comprende 7070 caratteri che possono essere raggruppati nel modo seguente.

- Caratteri a byte-singolo JIS X 0201 (191 caratteri, alias caratteri a mezza larghezza)
- Caratteri a doppio-byte JIS X 0208 (6879 caratteri, alias caratteri a piena larghezza)

I caratteri a doppio-byte occupano, nei terminali console che usano tipi di carattere CJK a spaziatura fissa, una larghezza doppia. Per gestire tale situazione, possono essere usati i file [HBF \(Hanzi Bitmap Font\)](#), con estensione ".hbf", per avere tipi di carattere contententi caratteri a byte-singolo e doppio-byte.

Per risparmiare spazio, per i file di caratteri [TrueType](#), si possono usare file con raccolte di tipi di carattere [TrueType](#) con estensione ".ttc".

Per gestire complesse spaziature dei caratteri, i tipi di carattere [PostScript](#) CID Type 1 sono usati con file CMap che iniziano con "%!PS-Adobe-3.0 Resource-CMap". Ciò viene raramente usato per il normale display X, ma è usato per la resa in file PDF, ecc. (vedere Sezione [7.7.2](#)).

Suggerimento

Per alcuni punti della codifica [Unicode](#) sono presenti [glifi](#) multipli, a causa della [unificazione Han](#). Un esmpio di glifi più problematici sono "U+3001 IDEOGRAPHIC COMMA" e "U+3002 IDEOGRAPHIC FULL STOP" la cui posizione è diversa tra le nazioni CJK. La configurazione della priorità dei caratteri pensati per il giapponese, rispetto a quelli cinesi, usando "~/.fonts.conf" dovrebbe dare un po' di pace agli utenti giapponesi.

7.7 Applicazioni per X

7.7.1 Applicazioni da ufficio per X

Ecco un elenco di applicazioni base per l'ufficio (LO è LibreOffice).

pacchetto	popcon	dimensione del pac- chetto	tipo	descrizione
libreoffice-writer	V:344, I:483	30505	LO	elaboratore di testi
libreoffice-calc	V:339, I:478	27827	LO	foglio di calcolo
libreoffice-impress	V:336, I:474	3807	LO	presentazioni
libreoffice-base	V:325, I:458	8883	LO	gestione di database
libreoffice-draw	V:337, I:475	9762	LO	editor di grafica vettoriale (draw)
libreoffice-math	V:340, I:480	1589	LO	editor di equazioni matematiche e formule
abiword	V:4, I:17	4748	GNOME	elaboratore di testi
gnumeric	V:16, I:25	7888	GNOME	foglio di calcolo
gimp	V:61, I:497	16421	GTK	editor di grafica bitmap (paint)
inkscape	V:165, I:340	129802	GNOME	editor di grafica vettoriale (draw)
dia	V:25, I:38	3910	GTK	editor di grafi e diagrammi di flusso
planner	V:4, I:7	1170	GNOME	gestione progetti
calligrawords	V:0, I:8	5634	KDE	elaboratore di testi
calligrasheets	V:0, I:7	13784	KDE	foglio di calcolo
calligrastage	V:0, I:6	4893	KDE	presentazioni
calligraplan	V:0, I:7	8178	KDE	gestione progetti
kexi	V:0, I:7	7391	KDE	gestione di database
karbon	V:0, I:8	3192	KDE	editor di grafica vettoriale (draw)

Tabella 7.8: Elenco di applicazioni base da ufficio per X

7.7.2 Applicazioni di utilità per X

Ecco un elenco di utilità base che hanno attirato la mia attenzione.



Attenzione

Per far sì che evince e okular visualizzino documenti PDF CJK usando dati Cmap (Sezione [7.6.3](#)), deve essere installato il pacchetto `poppler-data` (precedentemente non libero, vedere Sezione [11.4.1](#)).

Nota

L'installazione di software come `scribus` (KDE) nell'ambiente desktop GNOME è piuttosto accettabile dato che le funzionalità corrispondenti non sono disponibili in GNOME. Tuttavia l'installazione di troppi pacchetti con le stesse funzionalità ingombra il menu.

pacchetto	popcon	dimensione del pac- chetto	tipo	descrizione
evince	V:174, I:397	1406	GNOME	visualizzatore di documenti (pdf)
okular	V:77, I:122	13257	KDE	visualizzatore di documenti (pdf)
calibre	V:8, I:38	86119	KDE	convertitore e gestione di una biblioteca per libri elettronici
fbreader	V:2, I:17	3035	GTK	Lettori di ebook
evolution	V:40, I:280	440	GNOME	gestione di informazioni personali (groupware e posta elettronica)
kontact	V:2, I:23	2030	KDE	gestione di informazioni personali (groupware e posta elettronica)
scribus	V:6, I:27	19061	KDE	editor di impaginazione per il desktop
glabels	V:0, I:5	1389	GNOME	editor di etichette
gnucash	V:4, I:14	9667	GNOME	contabilità personale
homebank	V:0, I:3	878	GTK	contabilità personale
kmy money	V:0, I:3	12966	KDE	contabilità personale
shotwell	V:20, I:194	6635	GTK	organizzatore di foto digitali
xsane	V:21, I:192	909	GTK	frontend per lo scanner

Tabella 7.9: Elenco di applicazioni base di utilità per X

7.8 Informazioni varie su X

7.8.1 Appunti

La selezione in X usando il terzo pulsante del mouse è la funzionalità per appunti nativa di X (vedere Sezione [1.4.4](#)).

Suggerimento

Maiusc-Insert può funzionare come equivalente del clic del pulsante centrale del mouse.

pacchetto	popcon	dimensione del pac- chetto	tipo	descrizione
xsel	V:11, I:41	51	per X	Interfaccia a riga di comando per la selezione in X
xclip	V:8, I:37	55	per X	Interfaccia a riga di comando per la selezione in X

Tabella 7.10: Elenco di programmi di base per selezione in X

Gli ambienti desktop moderni (GNOME, KDE, ...) offrono sistemi diversi per gli appunti per le azioni di taglio, copia e incolla, utilizzando il pulsante sinistro del mouse e i tasti (Ctrl-x, Ctrl-c e Ctrl-v).

7.8.2 Mappatura della tastiera e dei pulsanti del dispositivo di puntamento in X

`xmodmap(1)` è un'utilità per modificare mappature di tastiera e dei pulsanti del dispositivo di puntamento nel sistema X Window. Per ottenere il **codice dei tasti**, eseguire `xev(1)` in X e premere i tasti. Per capire il significato di **keysym**, guardare la definizione di MACRO nel file `"/usr/include/X11/keysymdef.h"` (pacchetto `x11proto-core-dev`). Tutte le dichiarazioni `"#define"` in questo file hanno un nome che inizia con `"XK_"` che è seguito da i nomi **keysym**.

7.8.3 Client X classici

La maggior parte dei programmi client X tradizionali, come `xterm(1)`, può essere avviata con un insieme di opzioni a riga di comando standard per specificare la geometria, il tipo di carattere e il display.

Usano anche il database delle risorse X per configurare il loro aspetto. I valori predefiniti a livello di sistema delle risorse X sono memorizzati in `"/etc/X11/Xresources/*"` e i valori predefiniti per le singole applicazioni sono memorizzati in `"/etc/X11/app-defaults/*"`. Usare queste impostazioni come punto di partenza.

Il file `~/.Xresources` viene usato per memorizzare valori specifici dell'utente per le risorse. Questo file viene automaticamente unito ai valori predefiniti delle risorse X al momento del login. Per fare cambiamenti a queste impostazioni e rendere attivo immediatamente il loro effetto, unire le loro impostazioni al database usando il comando seguente.

```
$ xrbdb -merge ~/.Xresources
```

Vedere `x(7)` e `xrbdb(1)`.

7.8.4 L'emulatore di terminale per X - xterm

Si può imparare tutto ciò che c'è da sapere su `xterm(1)` all'indirizzo <http://dickey.his.com/xterm/xterm.faq.html>.

7.8.5 Eseguire client X come root

**avvertimento**

Non avviare mai il gestore di display/sessione X dall'account root digitando `"root"` al prompt di un display manager come `gdm3`, perché è considerato rischioso (non sicuro), anche quando si ha in mente di eseguire attività di amministrazione del sistema. L'intera architettura X è considerata non sicura se eseguita come utente root. Bisogna sempre utilizzare il livello di privilegi più basso possibile, come un account di un utente regolare.

Un modo facile di eseguire un particolare client X, ad esempio `"pippo"`, come root è l'uso di `sudo(8)` ecc., nel modo seguente.

```
$ sudo pippo &
```

```
$ sudo -s  
# pippo &
```

```
$ gksu pippo &
```

```
$ ssh -X root@localhost  
# pippo &
```

**Attenzione**

L'uso di `ssh(1)` solo per questo scopo, come nell'esempio precedente, è uno spreco di risorse.

Per fare sì che il client X si connetta al server X, tenere a mente quanto segue.

- I valori delle variabili d'ambiente `"$XAUTHORITY"` e `"$DISPLAY"` del vecchio utente devono essere copiati nelle variabili del nuovo utente.

- Il nuovo utente deve avere i permessi di lettura per il file a cui punta la variabile d'ambiente "\$XAUTHORITY".

Il pacchetto **gksu** (popcon: V:41, I:259) è un pacchetto specializzato, con interfaccia utente grafica GTK+, per ottenere i privilegi di root. Può essere configurato per usare `su(1)` o `sudo(8)` come backend, a seconda della chiave di gconf `"/apps/gksu/sudo-mode"`. Si può modificare la chiave di gconf usando `gconf-editor(1)` (menu: "Applicazioni" → "Strumenti di sistema" → "Editor della configurazione").

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

Ci sono 17, 18 o 10 lettere tra le lettere "m" e "n", "i" e "n" o "l" e "n" in, rispettivamente, "multilingualization", "internazionalization" e "localization" che sono i termini inglesi corrispondenti a M17N, I18N e L10N.

Il software moderno, come GNOME e KDE ha il supporto per le lingue native. È internazionalizzato rendendolo capace di gestire dati [UTF-8](#) e localizzato fornendo i messaggi tradotti attraverso l'infrastruttura `gettext(1)`. I messaggi tradotti possono essere forniti in pacchetti separati di localizzazione e possono essere selezionati semplicemente impostando al valore di localizzazione appropriato le variabili d'ambiente pertinenti.

La più semplice rappresentazione di dati testuali è **ASCII** che è sufficiente per l'inglese ed usa meno di 127 caratteri (rappresentabili con 7 bit). Al fine di supportare molti più caratteri per il supporto internazionale, sono stati inventati molti sistemi di codifica dei caratteri. Il sistema di codifica moderno e intelligente è **UTF-8** che può in pratica gestire tutti i caratteri conosciuti (vedere Sezione [8.4.1](#)).

Vedere [Introduzione all'i18n](#) per i dettagli.

Il supporto internazionale per l'hardware viene abilitato con dati di configurazione dell'hardware localizzati.



avvertimento

Questo capitolo sta diventando non aggiornato dato che è basato su Debian 7.0 (wheezy), rilasciata nel 2013.

8.1 L'input da tastiera

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
```

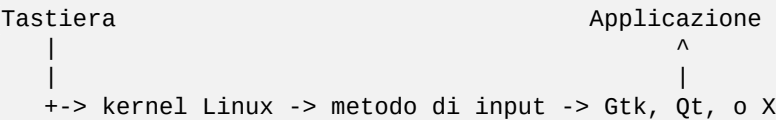
Ciò configura la tastiera per la console Linux e i parametri di configurazione di X Windows in `"/etc/default/keyboard"` e `"/etc/default/console-setup"`. Configura anche il tipo di carattere per la console Linux.

Molti caratteri non ASCII, inclusi quelli accentati usati da molte lingue europee, possono essere resi disponibili con il ["tasto morto"](#) (dead key), il [tasto AltGr](#) e il [tasto di composizione](#) (compose).

Per le lingue asiatiche è necessario il supporto per [metodi di input](#) più complessi, come [IBus](#) che sarà trattato in seguito.

8.1.1 Il supporto per metodo di input con IBus

L'input multilingua verso l'applicazione è elaborato così:



La configurazione dell'input per più lingue per il sistema Debian è semplificato dall'uso della famiglia di pacchetti [IBus](#) con il pacchetto `im-config`. Quello che segue è un elenco dei pacchetti IBus.

pacchetto	popcon	dimensione	localizzazioni supportate
ibus	V:6, I:10	42508	infrastruttura per metodo di input che usa dbus
ibus-mozc	V:1, I:2	927	giapponese
ibus-anthy	V:1, I:2	8450	" "
ibus-kkc	V:0, I:0	205	" "
ibus-skk	V:0, I:0	234	" "
ibus-pinyin	V:0, I:2	1425	cinese (per zh_CN)
ibus-chewing	V:0, I:0	401	" " (per zh_TW)
ibus-hangul	V:0, I:1	206	coreano
ibus-table	V:0, I:1	908	motore di tabelle per IBus
ibus-table-thai	I:0	45	thailandese
ibus-unikey	V:0, I:0	256	vietnamita
ibus-m17n	V:0, I:0	121	multilingue: indico, arabico e altri

Tabella 8.1: Elenco dei metodi di input supportati da IBus

Il metodo `kinput2` e altri classici [metodi di input](#) dipendenti dalla localizzazione per le lingue asiatiche esistono sempre, ma non sono raccomandati per i moderni ambienti X UTF-8. Gli insiemi di strumenti [SCIM](#) e [uim](#) costituiscono un approccio un po' più datato al metodo di input internazionale per gli ambienti X UTF-8 moderni.

8.1.2 Un esempio per il giapponese

Trovo che il metodo di input per il giapponese avviato dall'ambiente inglese (`"en_US.UTF-8"`) sia molto utile. Ecco come farlo con IBus per GNOME3:

1. Installare il pacchetto con gli strumenti di input per il giapponese `ibus-anthy` insieme ai pacchetti che raccomanda, come `im-config`.
2. Eseguire `"im-config"` da una shell utente e selezionare `"ibus"` come metodo di input.
3. Selezionare `"Settings"` → `"Keyboard"` → `"Input Sources"` → fare clic su `"+"` in `"Input Sources"` → `"Japanese"` → `"Japanese (anthy)"` e fare clic su `"Add"`.
4. Selezionare `"Japanese"` e fare clic su `"Add"` per supportare la disposizione giapponese di tastiera senza conversione di caratteri. (Si possono scegliere più fonti di input.)
5. Rieseguire il login nell'account utente

6. Verificare le impostazioni con `im-config`.
7. Impostare la fonte di input cliccando con il pulsante destro sull'icona della barra degli strumenti GUI.
8. Passare da una all'altra delle fonti di input installate con SUPER-SPAZIO. (SUPER è solitamente il tasto Windows.)

Notare quanto segue.

- `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.
- La voce di menu GUI per `im-config(8)` è disabilitata in modo predefinito per evitare sovraffollamenti.

8.1.3 Disabilitare il metodo di input

Se si desidera inserire dell'input senza passare da XIM (meccanismo usato da X), impostare il valore di `"$XMODIFIERS"` a `"none"` quando si avvia un programma. Una situazione di questo tipo potrebbe verificarsi quando si usa l'infrastruttura di input in giapponese `egg` in `emacs(1)` mentre è disabilitato `ibus`. Dalla shell usare il comando seguente.

```
$ XMODIFIERS=none emacs
```

Per modificare il comando eseguito dal menu Debian, posizionare una configurazione personalizzata in `"/etc/menu/"` seguendo il metodo descritto in `"/usr/share/doc/menu/html"`.

8.2 L'output sul display

La console Linux può mostrare solamente un numero limitato di caratteri. (È necessario usare speciali programmi per terminale come `jfbterm(1)` per mostrare lingue non Europee nella console non-X.)

X Window può mostrare qualsiasi carattere in UTF-8 purché esistano i tipi di carattere richiesti. (La codifica dei dati dei tipi di carattere originali viene gestita dal sistema X Window in modo trasparente per l'utente.)

8.3 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`: Modifica → Preferenze → Profili → Modifica → Compatibilità → Caratteri a larghezza ambigua → Stretto
- `ncurses`: impostare l'ambiente `export NCURSES_NO_UTF8_ACS=0`.

8.4 La localizzazione

Questa sezione si concentra sulla localizzazione per applicazioni eseguite nell'ambiente X Window avviato da `gdm3(1)`.

8.4.1 Nozioni base sulla codifica

La variabile d'ambiente `"LANG=xx_YY.ZZZZ"` imposta la localizzazione al codice di lingua `"xx"`, alla codice di nazione `"yy"` e alla codifica `"ZZZZ"` (vedere Sezione 1.5.2).

Il sistema Debian attuale normalmente imposta la localizzazione come `"LANG=xx_YY.UTF-8"` che fa uso della codifica [UTF-8](#) con l'insieme di caratteri [Unicode](#). Il sistema di codifica [UTF-8](#) è un sistema multibyte e usa i punti del codice in modo intelligente. I dati [ASCII](#), che consistono solamente di codici a 7 bit, sono comunque dati UTF-8 validi che usano 1 solo byte per carattere.

I sistemi Debian precedenti impostavano la localizzazione come `"LANG=C"` o `"LANG=xx_YY"` (senza `".UTF-8"`).

- Per `"LANG=C"` o `"LANG=POSIX"` viene usato l'insieme di caratteri [ASCII](#).
- Il sistema di codifica tradizionale in Unix è usato per `"LANG=xx_YY"`.

I sistemi di codifica tradizionale di fatto usati per `"LANG=xx_YY"` possono essere identificati controllando `"/usr/share/i18n/SUPP"`. Per esempio, `"en_US"` usa la codifica `ISO-8859-1` e `"it_IT@euro"` usa la codifica `ISO-8859-15`.

Suggerimento

Per il significato dei valori delle codifiche vedere Tabella 11.2.

8.4.2 Logica alla base dell'uso della localizzazione UTF-8

L'insieme di caratteri [Unicode](#) può rappresentare praticamente tutti i caratteri conosciuti con un intervallo di codici da 0 a 10FFFF in notazione esadecimale. La sua memorizzazione richiede almeno 21 bit.

Il sistema di codifica [UTF-8](#) fa rientrare i codici Unicode in un flusso di dati ragionevole a 8 bit compatibile con il sistema di elaborazione dei dati ASCII. **UTF** sta per Unicode Transformation Format (formato di trasformazione di Unicode).

È raccomandato l'uso della localizzazione [UTF-8](#) per il desktop, per esempio `"LANG=en_US.UTF-8"`. La prima parte della localizzazione determina i messaggi presentati dalle applicazioni. Per esempio, `gedit(1)` (un editor di testi per il desktop GNOME) nella localizzazione `"LANG=it_IT.UTF-8"` può mostrare e modificare testi in caratteri cinesi e allo stesso tempo presentare i menu in italiano, a patto che siano installati i necessari tipi di carattere e metodi di input.

È raccomandabile anche impostare la localizzazione usando solamente la variabile d'ambiente `"$LANG"`. Non vedo la convenienza nell'impostare nella localizzazione UTF-8 una complessa combinazione di variabili `"LC_*"` (vedere `locale(1)`).

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.

```
"citazione tra virgolette doppie" non è uguale a "citazione tra virgolette doppie ASCII"
'citazione tra virgolette singole' non è uguale a 'citazione tra virgolette singole ASCII'
```

Quando dati di testo in puro [ASCII](#) sono convertiti in [UTF-8](#) hanno contenuto e dimensioni esattamente identici a quelli dei dati ASCII originali. Perciò non si può perdere nulla usando la localizzazione UTF-8.

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.

I vecchi sistemi di codifica non UTF-8 specifici di produttori, tendono ad avere piccolissime, ma noiose, differenze per molte nazioni in alcuni caratteri come quelli grafici. L'uso del sistema UTF-8 da parte dei sistemi operativi moderni ha praticamente risolto questi problemi di conflitti tra le codifiche.

8.4.3 La (ri)configurazione della localizzazione

Per far sì che un sistema abbia accesso ad una localizzazione particolare, è necessario che i dati della localizzazione siano stati compilati a partire dal database della localizzazione. (Il sistema Debian **non** viene fornito con tutte le localizzazioni disponibili già compilate a meno che non sia stato installato il pacchetto `locales-all`.) L'elenco completo delle localizzazioni supportate disponibili per la compilazione è disponibile in `/usr/share/i18n/SUPPORTED` che elenca tutti i nomi esatti delle localizzazioni. Il comando seguente elenca tutte le localizzazioni UTF-8 già compilate in forma binaria.

```
$ locale -a | grep utf8
```

Si può riconfigurare il pacchetto `locales` eseguendo il comando seguente.

```
# dpkg-reconfigure locales
```

Questo processo comprende 3 passi successivi.

1. Aggiornare l'elenco delle localizzazioni disponibili
2. Compilarle in forma binaria
3. Impostare, per l'uso da parte di PAM (vedere Sezione 4.5) il valore della localizzazione di sistema nel file `/etc/default/locale`

L'elenco delle localizzazioni disponibili dovrebbe includere `en_US.UTF-8` e tutte le lingue desiderate con `UTF-8`.

La localizzazione predefinita raccomandata è `en_US.UTF-8` per l'inglese statunitense. Per le altre lingue, assicurarsi di scegliere la localizzazione con `UTF-8`; ognuna di queste può gestire qualsiasi carattere internazionale.

Nota

Benché l'impostazione della localizzazione a `C` usi i messaggi in inglese statunitense, gestisce solo i caratteri ASCII.

8.4.4 Il valore della variabile di ambiente `$LANG`

Il valore della variabile d'ambiente `$LANG` viene impostato e modificato da molte applicazioni.

- È impostato inizialmente dal meccanismo PAM di `login(1)` per i programmi nelle console Linux locali
- È impostato inizialmente dal meccanismo PAM del display manager per tutti i programmi X
- È impostato inizialmente dal meccanismo PAM per `ssh(1)` per i programmi nelle console remote
- È modificato da alcuni display manager come `gdm3(1)` per tutti i programmi X
- È modificato dal codice di avvio della sessione X tramite `~/ .xsessionrc` per tutti i programmi X
- È modificato dal codice di avvio della shell, ad esempio `~/ .bashrc` per tutti i programmi in console

Suggerimento

È una buona idea installare la localizzazione predefinita per tutto il sistema come `en_US.UTF-8` per avere la massima compatibilità.

8.4.5 Localizzazione specifica solamente in X Window

Si può scegliere una localizzazione specifica solo sotto X Window, indipendentemente dalla localizzazione predefinita di sistema, usando la personalizzazione di PAM (vedere Sezione 4.5) nel modo descritto in seguito.

Questo ambiente dovrebbe fornire la migliore esperienza desktop possibile mantenendo la stabilità. Si ha accesso ad un terminale a caratteri funzionante con messaggi leggibili, anche quando il sistema X Window non funziona. Ciò diventa essenziale per le lingue che usano caratteri non romani, come cinese, giapponese e coreano.

Nota

Potrebbe essere disponibile un modo alternativo per ottenere lo stesso risultato, visti i miglioramenti che avvengono nei pacchetti dei gestori delle sessioni X, ma la lettura di quanto segue fornisce un metodo generico di base per l'impostazione della localizzazione. Per gdm3(1), è possibile selezionare la localizzazione della sessione X dal suo menu.

La riga seguente definisce la posizione del file dell'ambiente della lingua, nel file di configurazione di PAM, come `/etc/pam.d/gdm3`

```
auth    required      pam_env.so read_env=1 envfile=/etc/default/locale
```

Cambiare tale riga nel modo seguente

```
auth    required      pam_env.so read_env=1 envfile=/etc/default/locale-x
```

Per il giapponese, creare un file `/etc/default/locale-x` permessi `-rw-r--r-- 1 root root` contenente quanto segue.

```
LANG="ja_JP.UTF-8"
```

Mantenere il file predefinito `/etc/default/locale` per l'uso da parte di altri programmi nel modo seguente.

```
LANG="en_US.UTF-8"
```

Questa è la tecnica più generica per personalizzare la localizzazione e rende localizzato il dialogo di selezione stesso del menu di gdm3(1).

In alternativa in questo caso, si può semplicemente cambiare la localizzazione usando il file `~/.xsessionrc`.

8.4.6 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.

8.4.7 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 <nomeprogramma>
```

8.4.8 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 [9.2.5](#)).

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.

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 Il programma screen

screen(1) è uno strumento molto utile per permettere alle persone di accedere a siti remoti attraverso una connessione non affidabile o intermittente, dato che supporta connessioni interrotte.

pacchetto	popcon	dimensione	descrizione
screen	V:186, I:292	987	multiplexer per terminale con emulazione di terminale VT100/ANSI

Tabella 9.1: Elenco di programmi che supportano connessioni di rete interrotte

9.1.1 Scenario di uso per screen(1)

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.

9.1.2 Associazioni dei tasti per il comando screen

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.

azione associata	significato
^A ?	mostra una schermata di aiuto (visualizza le associazioni di tasti)
^A c	crea una nuova finestra e si sposta in essa
^A n	va alla finestra successiva
^A p	va alla finestra precedente
^A 0	va alla finestra numero 0
^A 1	va alla finestra numero 1
^A w	mostra l'elenco delle finestre
^A a	invia Ctrl-A alla finestra attuale come input da tastiera
^A h	scrive una copia della schermata della finestra attuale in un file
^A H	inizia/termina la registrazione dell'attività della finestra attuale in un file
^A ^X	blocca il terminale (protetto da password)
^A d	scollega la sessione screen dal terminale
^A DD	scollega la sessione di screen e fa il log out

Tabella 9.2: Elenco di associazioni di tasti per screen

Vedere screen(1) per i dettagli.

9.2 Registrazione e presentazione di dati

9.2.1 Demoni di registro

Molti programmi registrano le proprie attività nella directory `"/var/log/"`.

- Il demone del registro del kernel: klogd(8)
- Il demone del registro di sistema: rsyslogd(8)

Vedere Sezione 3.2.9 e Sezione 3.2.10.

9.2.2 Analizzatori di registro

Quelli che seguono sono alcuni analizzatori di registro degni di nota ("`~Gsecurity::log-analyzer`" in aptitude(8)).

pacchetto	popcon	dimensione	descrizione
logwatch	V:18, I:21	2214	analizzatore di registro scritto in Perl con un bell'output
fail2ban	V:100, I:111	1302	strumento per interdire IP che causano errori di autenticazione multipli
analog	V:5, I:125	3529	analizzatore del registro di server web
awstats	V:12, I:19	6799	analizzatore potente e ricco di funzionalità per registro di server web
sarg	V:5, I:5	432	generatore di rapporti sull'analisi di squid
pflogsumm	V:1, I:4	111	strumento per riassunti delle voci di registro di Postfix
syslog-summary	V:1, I:4	30	riassume i contenuti di un file di registro di sistema
fwlogwatch	V:0, I:0	474	analizzatore del registro del firewall
squidview	V:0, I:2	192	monitora e analizza i file access.log di squid
swatch	V:0, I:0	100	visualizzatore di file di registro con corrispondenze con espressioni regolari, evidenziazione ed eventi
crm114	V:0, I:1	1099	Controllable Regex Mutilator e filtro per spam (CRM114)
icmpinfo	V:0, I:0	39	interpreta i messaggi ICMP

Tabella 9.3: Elenco di analizzatori del registro di sistema

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.

9.2.3 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 è avviato, il file è typescript
```

Fare tutto quello che si vuole ...e poi premere **Ctrl-D** per uscire da `script`.

```
$ col -bx <typescript >filepulito
$ vim filepulito
```

Se `script` non è disponibile (per esempio, durante il processo di avvio in `initramfs`), si può usare invece il comando seguente.

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

Suggerimento

Alcuni `x-terminal-emulator` (emulatori di terminale in X), come `Terminale di GNOME` hanno la funzione di registrazione. Si potrebbe voler ampliare il buffer delle righe per scorrerle all'indietro.

Suggerimento

Si può usare `screen(1)` con `"^A H"` (vedere Sezione [9.1.2](#)) per registrare la console.

Suggerimento

È possibile usare `emacs(1)` con `"M-x shell"`, `"M-x eshell"` o `"M-x term"` per registrare la console. Si può successivamente usare `"C-x C-w"` per scrivere il buffer in un file.

9.2.4 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.

9.2.5 Visualizzazione personalizzata di date e orari

Il formato predefinito per la visualizzazione delle date e degli orari per il comando `"ls -l"` dipende dalla **localizzazione** (vedere Sezione 1.2.6 per il valore). Si fa dapprima riferimento alla variabile `"$LANG"` che può essere scavalcata dalla variabile `"$LC_TIME"`.

Il formato effettivo di visualizzazione predefinito per ciascuna localizzazione dipende dalla versione della libreria C (il pacchetto `libc6`) usata. Differenti rilasci di Debian hanno cioè valori predefiniti diversi.

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 9.4: Esempi di visualizzazione di date e orari per il comando `"ls -l"` per wheezy

Suggerimento

Si può evitare di digitare lunghe opzioni nella riga di comando usando alias, per esempio `"alias ls='ls --time-style=+%d.%m.%y\ %H:%M'"` (vedere Sezione 1.5.9).

Suggerimento

Per i formati iso viene seguito lo standard [ISO 8601](#).

9.2.6 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 "${RED}TESTO-ROSSO${NORMAL} ${REVERSE}TESTO-IN-NEGATIVO${NORMAL}"
```

9.2.7 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"`.

9.2.8 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.
-

9.2.9 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	comando
xbase-clients	I:53	46	<code>xwd(1)</code>
gimp	V:61, I:497	16421	menu della GUI
imagemagick	V:77, I:551	196	<code>import(1)</code>
scrot	V:9, I:91	50	<code>scrot(1)</code>

Tabella 9.5: Elenco di strumenti di manipolazione di immagini grafiche

9.2.10 Registrare i cambiamenti a file di configurazione

Esistono strumenti specializzati per registrare i cambiamenti nei file di configurazione con l'aiuto del sistema DVCS.

pacchetto	popcon	dimensione	descrizione
etckeeper	V:25, I:29	153	archivia i file di configurazione ed i loro metadati con Git (predefinito), Mercurial o Bazaar (nuovo)
changetrack	V:0, I:0	62	archivia i file di configurazione con RCS (vecchio)

Tabella 9.6: Elenco di pacchetti per registrare la cronologia della configurazione in VCS

Io raccomando l'uso del pacchetto `etckeeper` con `git(1)` che mette l'intera directory `"/etc"` sotto il controllo di VCS. La sua guida di installazione ed il suo tutorial si trovano in `"/usr/share/doc/etckeeper/README.gz"`.

Fondamentalmente, l'esecuzione di `"sudo etckeeper init"` inizializza il repository `git` per `"/etc"`, esattamente come nel procedimento spiegato in Sezione 10.6.5 ma con speciali script attivati da eventi per impostazioni più accurate.

Quando si cambia la configurazione, si può usare normalmente `git(1)` per registrare i cambiamenti. Esso registra automaticamente i cambiamenti in modo comodo anche ogni volta che vengono eseguiti i comandi di gestione dei pacchetti.

Suggerimento

Si può sfogliare la cronologia dei cambiamenti di `"/etc"` eseguendo `"sudo GIT_DIR=/etc/.git"` ottenendo una chiara visualizzazione dei nuovi pacchetti installati, di quelli rimossi e dei cambiamenti di versione dei pacchetti.

9.3 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.

9.3.1 Cronometrare un processo

Mostrare il tempo usato dal processo invocato da un comando.

```
# time un_qualche_comando >/dev/null
real    0m0.035s    # tempo di orologio (tempo reale trascorso)
user    0m0.000s    # tempo in modalità utente
sys     0m0.020s    # tempo in modalità kernel
```

pacchetto	popcon	dimensione	descrizione
coreutils	V:886, I:999	15016	nice(1): esegue un programma con priorità di schedulazione modificata
bsdutils	V:860, I:999	255	renice(1): modifica la priorità di schedulazione di un processo in esecuzione
procps	V:818, I:999	690	utilità per il file system <code>"/proc"</code> : ps(1), top(1), kill(1), watch(1), ...
psmisc	V:373, I:934	526	utilità per il file system <code>"/proc"</code> : killall(1), fuser(1), peekfd(1), pstree(1)
time	V:29, I:606	82	time(1): esegue un programma per riportare l'uso delle risorse di sistema in funzione del tempo
sysstat	V:123, I:145	1454	sar(1), iostat(1), mpstat(1), ...: strumenti per le prestazioni di sistema per Linux
isag	V:0, I:4	109	Interactive System Activity Grapher (tracciamento interattivo dell'attività) per sysstat
lsof	V:187, I:947	440	lsof(8): elenca, usando l'opzione <code>"-p"</code> , i file aperti da un processo in esecuzione
strace	V:21, I:163	1606	strace(1): traccia le chiamate e i segnali di sistema
ltrace	V:1, I:22	360	ltrace(1): traccia le chiamate di libreria
xtrace	V:0, I:1	333	xtrace(1): traccia la comunicazione tra client e server X11
powertop	V:6, I:231	612	powertop(1): informazioni sull'uso dell'alimentazione da parte del sistema
cron	V:882, I:997	251	esegue processi dal demone cron(8) sullo sfondo in base ad una pianificazione
anacron	V:449, I:528	106	pianificatore di comandi in stile cron, per i sistemi che non sono in esecuzione 24 ore al giorno
at	V:364, I:621	149	at(1) o batch(1): esegue un compito ad un orario specificato o quando il carico di sistema scende sotto un certo livello

Tabella 9.7: Elenco di strumenti per monitorare e controllare l'attività dei programmi.

9.3.2 Priorità di schedulazione

Per controllare la priorità di schedulazione di un processo è usato il valore di nice.

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 9.8: Elenco di valori di nice per la priorità di schedulazione

```
# nice -19 top # molto gentile
# nice --20 wodim -v -eject speed=2 dev=0,0 disk.img # molto veloce
```

A volte un valore molto alto di nice fa più male che bene al sistema; usare quindi questo comando con molta cautela.

9.3.3 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	ps aux	mostra %CPU %MEM
System V	ps -efH	mostra PPID

Tabella 9.9: 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.

9.3.4 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».

9.3.5 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`.

9.3.6 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
```

9.3.7 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).

9.3.8 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.

9.3.9 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 "$x" ]; then comando "$x" ; fi; done
```

- Combinazione di `find(1)` e `xargs(1)`:

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | xargs -0 -n 1 comando
```

- `find(1)` con l'opzione `"-exec"` con un comando:

```
find . -type f -maxdepth 1 -name '*.ext' -exec comando '{}' \;
```

- `find(1)` con l'opzione `"-exec"` con un breve script di shell:

```
find . -type f -maxdepth 1 -name '*.ext' -exec sh -c "comando '{}'" && echo 'esecuzione con ←
successo'" \;
```

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](#).

9.3.10 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](#).

Per l'[interfaccia utente grafica \(GUI\)](#) conforme agli standard di freedesktop.org i file *.desktop nella directory /usr/share/applications forniscono gli attributi necessari per la visualizzazione di ogni programma nel menu della GUI. Vedere Sezione [7.2.2](#).

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 Specifica 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[it]=Esplorare il 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)`.

9.3.11 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	funzione
1	HUP	riavvia il demone
15	TERM	normale uccisione
9	KILL	uccisione forzata

Tabella 9.10: Elenco dei segnali usati comunemente con il comando kill

9.3.12 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).

9.3.13 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 'comando -argomenti' | at 3:40 monday
```

9.3.14 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).

```
# usare /bin/sh per eseguire i comandi, qualsiasi cosa dica /etc/passwd
SHELL=/bin/sh
# inviare per posta l'output a paolo, chiunque sia il proprietario di questo crontab
MAILTO=paolo
# Min Ora GiornoMese Mese GiornoSett comando (Giorni... sono combinati con OR)
# eseguito alle 00:05, ogni giorno
5 0 * * * $HOME/bin/compito.quotidiano >> $HOME/tmp/output 2>&1
# eseguito alle 14:15 il primo giorno di ogni mese -- output inviato a paolo
15 14 1 * * $HOME/bin/mensile
# eseguito alle 22:00 nei giorni infrasettimanali(1-5), disturbare Gianni. % per a capo, ←
ultimo % per cc:
0 22 * * 1-5 mail -s "Sono le 10pm" giannie%Gianni,%dove sono i tuoi ragazzi?%.%%
23 */2 2 * echo "eseguito 23 minuti dopo le 0am, 2am, 4am ..., il giorno 1 Feb "
5 4 * * sun echo "eseguito alle 04:05 ogni domenica"
# eseguito alle 03:40 il primo lunedì' di ogni mese
40 3 1-7 * * [ "$(date +%a)" == "Mon" ] && comando -opzioni
```

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`.

9.3.15 Tasto Alt-SysRq

L’opzione ”Magic SysRq key” ([tasto R Sist](#)), di compilazione del kernel, che è ora lo standard per i kernel Debian, fornisce una garanzia contro i malfunzionamenti del sistema. Premendo Alt-R Sist seguito da uno dei tasti seguenti, si ottiene magicamente di recuperare il controllo del sistema.

tasto premuto dopo Alt-R_Sist	descrizione dell’azione
r	ripristina la tastiera dalla modalità raw dopo un crash di X
0	cambia il livello di visualizzazione del registro sulla console a 0 per ridurre i messaggi di errore
k	uccide (kill) tutti i processi nella console virtuale attuale
e	invia un segnale SIGTERM a tutti i processi, tranne init(8)
i	invia un segnale SIGKILL a tutti i processi, tranne init(8)
s	sincronizza tutti i file system montati per evitare corruzione di dati
u	monta nuovamente tutti i file system in sola lettura (umount)
b	riavvia (reboot) il sistema senza sincronizzare o smontare i file system

Tabella 9.11: Elenco di tasti per il comando SAK

Suggerimento

Leggere le pagine di manuale signal(7), kill(1) e sync(1) per capire la descrizione soprastante.

La combinazione di ”Alt-SysRq s”, ”Alt-SysRq u” e ”Alt-SysRq r” è buona per uscire da situazioni veramente brutte e ottenere un accesso utilizzabile alla tastiera senza fermare il sistema.

Vedere ”/usr/share/doc/linux-doc-3.*/Documentation/sysrq.txt.gz”.



Attenzione

La funzione Alt-R_Sist può essere considerata un rischio per la sicurezza dato che permette agli utenti l’accesso a funzioni con privilegi di root. Per disabilitare la funzione Alt-R_Sist mettere ”echo 0 >/proc/sys/kernel/sysrq” in ”/etc/rc.local” o ”kernel.sysrq = 0” in ”/etc/sysctl.conf”.

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.

9.4 Suggerimenti per l’amministrazione del sistema

9.4.1 Chi è nel sistema?

Si può controllare chi è connesso al sistema nei modi seguenti.

- who(1) mostra chi ha fatto il login ed è connesso.
- w(1) mostra l’elenco di chi ha fatto il login ed è connesso e cosa sta facendo.
- last(1) mostra l’elenco degli utenti che hanno fatto il login più recentemente.

- `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)`.

9.4.2 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 "Arresto pianificato del sistema tra 1 ora" | wall
```

9.4.3 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:175, I:993	1220	utilità PCI Linux: <code>lspci(8)</code>
usbutils	V:104, I:862	721	utilità Linux USB: <code>lsusb(8)</code>
pcmciautils	V:22, I:42	94	utilità PCMCIA per Linux: <code>pccardctl(8)</code>
scsitools	V:0, I:4	366	raccosta di strumenti per la gestione dell’hardware SCSI: <code>lsscsi(8)</code>
procinfo	V:1, I:15	122	informazioni sul sistema ottenute da <code>"/proc"</code> : <code>lsdev(8)</code>
lshw	V:9, I:87	671	informazioni sulla configurazione hardware: <code>lshw(1)</code>
discover	V:40, I:934	90	sistema di identificazione dell’hardware: <code>discover(8)</code>

Tabella 9.12: Elenco di strumenti per l’identificazione dell’hardware

9.4.4 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.

[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`.

9.4.5 Orario di sistema e hardware

I comandi seguenti impostano l’ora di sistema e hardware a MM/GG hh:mm, AAAA.

```
# date MMGGhhmmAAAA
# hwclock --utc --systohc
# hwclock --show
```

pacchetto	popcon	dimensione	descrizione
console-setup	V:348, I:941	438	utilità Linux per i tipi di carattere e la mappa di tastiera in console
x11-xserver-utils	V:358, I:581	516	utilità del server X: xset(1), xmodmap(1)
acpid	V:382, I:675	143	demone per gestire eventi inviati dall'ACPI (Advanced Configuration and Power Interface, interfaccia avanzata per configurazione ed energia)
acpi	V:30, I:647	46	utilità per mostrare informazioni sui device ACPI
apmd	V:0, I:7	132	demone per gestire eventi inviati dall'APM (Advanced Power Management, gestione avanzata dell'energia)
sleepd	V:0, I:0	79	demone per sospendere un portatile durante l'inattività
hdparm	V:339, I:538	245	ottimizzazione degli accessi al disco fisso (vedere Sezione 9.5.9)
smartmontools	V:121, I:195	1716	controllo e monitoraggio dei sistemi di archiviazione usando S.M.A.R.T.
setserial	V:6, I:11	113	raccolta di strumenti per la gestione delle porte seriali
memtest86+	V:1, I:36	2391	raccolta di strumenti per la gestione di hardware di memoria
scsitools	V:0, I:4	366	raccolta di strumenti per la gestione di hardware SCSI
setcd	V:0, I:1	35	ottimizzazione dell'accesso alle unità CD
big-cursor	I:0	27	puntatori del mouse più grandi per X

Tabella 9.13: Elenco di strumenti di configurazione dell'hardware

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\(GMT\)](#).

Se l'ora hardware (BIOS) è impostata ad UTC, modificare l'impostazione nel file `"/etc/default/rcS"` ad `"UTC=yes"`.

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`, `ntpd` 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

`ntpdate(8)`, nel pacchetto `ntp` può tracciare una catena di server NTP all'indietro fino alla fonte originaria.

9.4.6 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 iedere `/usr/share/doc/libncurses5/FAQ`).

9.4.7 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 \(Open Sound System\)](#).

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.

I software applicativi possono essere configurati non solo per accedere direttamente a device audio, ma anche per accedervi attraverso un qualche sistema server sonoro standardizzato.

pacchetto	popcon	dimensione	descrizione
alsa-utils	V:388, I:526	2103	utilità per configurare ed usare ALSA
oss-compat	V:4, I:42	7	compatibilità OSS in ALSA per prevenire errori <code>"/dev/dsp not found"</code>
jackd	V:3, I:28	32	server JACK (JACK Audio Connection Kit) (bassa latenza)
libjack0	V:0, I:17	337	libreria JACK (JACK Audio Connection Kit) (bassa latenza)
nas	V:0, I:0	239	server NAS (Network Audio System)
libaudio2	V:46, I:551	161	libreria NAS (Network Audio System)
pulseaudio	V:370, I:501	6105	server PulseAudio , sostituto per ESD
libpulse0	V:246, I:648	989	libreria client PulseAudio , sostituto per ESD
libgststreamer1.0-0	V:243, I:513	4860	GStreamer : motore audio di GNOME
libphonon4	I:214	672	Phonon : motore audio di KDE

Tabella 9.14: Elenco di pacchetti relativi all'audio

Normalmente esiste un motore audio comune per ciascun ambiente desktop. Ciascun motore audio usato dalle applicazioni può scegliere di connettersi a diversi server audio.

9.4.8 Disabilitare il salvaschermo

Per disabilitare il salvaschermo usare i comandi seguenti.

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 9.15: Elenco di comandi per disabilitare il salvaschermo

9.4.9 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
```

9.4.10 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 9.16: Elenco di dimensioni della memoria riportate

9.4.11 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](#).
- [Manuale Securing Debian](#).

pacchetto	popcon	dimensione	descrizione
logcheck	V:11, I:14	152	demone per inviare via posta anomalie nei file di registro di sistema all'amministratore
debsums	V:6, I:40	120	utilità per controllare i file dei pacchetti installati con le somme di controllo MD5
chkrootkit	V:7, I:27	954	rilevatore di rootkit
clamav	V:16, I:69	893	utilità anti-virus per Unix - interfaccia a riga di comando
tiger	V:3, I:4	2599	riporta vulnerabilità nella sicurezza di sistema
tripwire	V:3, I:4	11480	strumento di controllo dell'integrità di file e directory
john	V:3, I:14	449	strumento di violazione delle password attive
aide	V:2, I:2	1998	AIDE - ambiente avanzato di rilevamento delle intrusioni - binario statico
integrit	V:0, I:0	313	programma di controllo dell'integrità di file
crack	V:0, I:0	128	programma per indovinare password

Tabella 9.17: Elenco di strumenti per verificare la sicurezza e l'integrità del sistema

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.

9.5 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.

9.5.1 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.

9.5.2 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.

La maggior parte dei PC usa il classico schema basato su [MBR \(Master Boot Record\)](#) per contenere i dati sul [partizionamento del disco](#) nel primo settore, cioè il settore [LBA 0](#) (512 byte).

Nota

Alcuni PC più recenti con [EFI \(Extensible Firmware Interface\)](#), compresi i Mac basati su Intel, usano lo schema [GPT \(GUID Partition Table\)](#) per contenere i dati sul [partizionamento del disco](#) non nel primo settore.

Sebbene `fdisk(8)` sia stato lo strumento standard per il partizionamento dei dischi, `parted(8)` lo sta sostituendo.

pacchetto	popcon	dimensione	GPT	descrizione
util-linux	V:887, I:999	3204	Non supportato	utilità di sistema varie inclusi <code>fdisk(8)</code> e <code>cdisk(8)</code>
parted	V:365, I:561	278	Supportato	programma GNU Parted per il ridimensionamento delle partizioni dei dischi
gparted	V:25, I:148	7231	Supportato	editor delle partizioni di GNOME basato su libparted
gdisk	V:56, I:493	779	Supportato	editor delle partizioni per i dischi GPT
kpartx	V:14, I:28	80	Supportato	programma per creare la mappatura a device per le partizioni

Tabella 9.18: 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 [9.7.6](#)) 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](#).

9.5.3 Accedere alle partizioni usando UUID

Anche se la riconfigurazione delle partizioni o l'ordine di attivazione di supporti di archiviazione removibili può portare ad avere nomi diversi per le partizioni, è possibile accedere ad esse in modo coerente. Ciò è utile anche se si hanno più dischi ed il BIOS non assegna loro un nome di dispositivo costante.

- `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)`.

Suggerimento

I nodi di device dei dispositivi come i supporti di archiviazione removibili possono essere resi statici usando [regole udev](#), se necessario. Vedere Sezione [3.3](#).

9.5.4 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

9.5.5 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](#)
 - `fsck.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 `unde1` per ripristinare file eliminati.)
-

pacchetto	popcon	dimensione	descrizione
e2fsprogs	V:487, I:999	1311	utilità per i file system ext2/ext3/ext4
reiserfsprogs	V:9, I:25	891	utilità per i file system Reiserfs
dosfstools	V:100, I:562	235	utilità per i file system FAT (Microsoft: MS-DOS, Windows)
xfsprogs	V:19, I:84	4338	utilità per i file system XFS (SGI: IRIX)
ntfs-3g	V:242, I:556	1355	utilità per i file system NTFS (Microsoft: Windows NT, ...)
jfsutils	V:1, I:13	1533	utilità per i file system JFS (IBM: AIX, OS/2)
reiser4progs	V:0, I:5	1325	utilità per i file system Reiser4
hfsprogs	V:0, I:9	303	utilità per i file system HFS e HFS Plus (Apple: Mac OS)
btrfs-tools	V:6, I:38	29	utilità per i file system Btrfs
zerofree	V:2, I:65	25	programma per impostare a 0 i blocchi liberi in file system ext2/3/4

Tabella 9.19: Elenco di pacchetti di gestione dei file system

I comandi `mkfs(8)` e `fsck(8)` sono forniti dal pacchetto `e2fsprogs` come front-end per vari programmi dipendenti dal file system (`mkfs.tipofs` e `fsck.tipofs`). Per il file system [ext4](#), sono `mkfs.ext4(8)` e `fsck.ext4(8)` (sono un collegamento simbolico a `mke2fs(8)` e `e2fsck(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 sia abbiano specifiche ragioni per non farlo.

Suggerimento

Il file system [Btrfs](#) è disponibile nel kernel Linux 3.2 (Debian wheezy). Dovrebbe diventare il prossimo file system predefinito dopo quello `ext4`.

**avvertimento**

Non si dovrebbe ancora usare il file system `Btrfs` per i propri dati importanti prima che abbia la funzionalità `fsck(8)` live in spazio kernel e il supporto nel bootloader.

Suggerimento

Alcuni strumenti permettono l'accesso a file system non supportati del kernel Linux (vedere Sezione [9.7.2](#)).

9.5.6 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)`.

Suggerimento

Per vedere i risultati del comando `fsck(8)` avviato dallo script di avvio, controllare i file in `"/var/log/fsck/"`.

9.5.7 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> <punto di mount> <tipo> <opzioni> <dump> <pass>
proc /proc proc defaults 0 0
UUID=709cbe4c-80c1-56db-8ab1-dbce3146d2f7 / ext4 noatime,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 9.5.3) invece del normale nome di device a blocchi quale `«/dev/sda1»`, `«/dev/sda2»`, ...

Le prestazioni e le caratteristiche di un file system possono essere ottimizzate usando per esso opzioni per il montaggio (vedere `fstab(5)` e `mount(8)`). Alcune opzioni importanti sono le seguenti.

- L'opzione `"defaults"` implica le opzioni predefinite: `"rw, suid, dev, exec, auto, nouser, async"`. (Uso generico)
- L'opzione `"noatime"` o `"relatime"` è molto efficace nel velocizzare l'accesso in lettura. (Uso generico)
- L'opzione `"user"` permette ad un normale utente di montare il file system. Questa opzione implica la combinazione di opzioni `"noexec, nosuid, nodev"`. (Uso generico, usato per CD o dispositivi di archiviazione USB)
- La combinazione di opzioni `"noexec, nodev, nosuid"` è usata per migliorare la sicurezza. (Uso generico)
- L'opzione `"noauto"` limita il montaggio solamente a quando esplicitamente richiesto. (Uso generico)
- L'opzione `"data=journal"` per `ext3fs` può migliorare la salvaguardia dell'integrità dei dati contro interruzioni dell'alimentazione a prezzo della perdita di un po' di velocità in scrittura.

Suggerimento

Per utilizzare una modalità di journaling non predefinita per il file system root, è necessario fornire i parametri di avvio del kernel (vedere Sezione 3.1.2), ad esempio `"rootflags=data=journal"`. Per lenny la modalità di journaling predefinita è `"rootflags=data=ordered"`; per squeeze è `"rootflags=data=writeback"`.

9.5.8 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](#).

9.5.9 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"`.

9.5.10 Ottimizzare le unità a stato solido

Le prestazioni e l'usura dei dischi delle [unità a stato solido \(SSD\)](#) possono essere ottimizzate nel modo seguente.

- Usare il kernel Linux più recente (≥ 3.2).
 - Ridurre le scritture su disco per gli accessi in lettura.
 - Impostare l'opzione di montaggio `"noatime"` o `"relatime"` in `/etc/fstab`.
 - Abilitare il comando [TRIM](#).
 - Impostare l'opzione di montaggio `"discard"` in `/etc/fstab` per il file system `ext4`, le partizioni di swap, Btrfs, ecc. Vedere `fstab(5)`.
 - Impostare l'opzione `"discard"` in `/etc/lvm/lvm.conf` per [LVM](#). Vedere `lvm.conf(5)`.
 - Impostare l'opzione `"discard"` in `/etc/crypttab` per [dm-crypt](#). Vedere `crypttab(5)`.
 - Abilitare lo schema di allocazione dello spazio su disco ottimizzato SSD.
 - Impostare l'opzione di montaggio `"ssd"` in `/etc/fstab` per Btrfs.
 - Fare sì che il sistema muova tutti i dati su disco ogni 10 minuti sui PC portatili.
 - Impostare l'opzione di montaggio `"commit=600"` in `/etc/fstab`. Vedere `fstab(5)`.
-

- Impostare pm-utils in modo che usi la modalità laptop-mode anche quando collegato alla rete elettrica. Vedere [Debian BTS #659260](#).

**avvertimento**

Modificare l'intervallo di scrittura dei dati dai normali 5 secondi a 10 minuti rende i dati vulnerabili in caso di mancanza di corrente.

9.5.11 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"`.
 - Riavviare il demone smartd(8) con il comando `"sudo /etc/init.d/smartmontools restart"`.

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.

9.5.12 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.

9.5.13 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.

9.5.14 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 dir-da-usare vecchia-dir
$ sudo mkfs.ext4 /dev/sdx
$ sudo mount -t ext4 /dev/sdx dir-da-usare
$ sudo cp -a vecchia-dir/* dir-da-usare
$ sudo rm -rf vecchia-dir
```

Suggerimento

In alternativa si può montare un file immagine vuoto del disco (vedere Sezione 9.6.5) come device loop (vedere Sezione 9.6.3). Il reale uso del disco cresce mano a mano che vengono archiviati i dati.

9.5.15 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 /percorso/della/dir-vuota dir-di-lavoro
```

9.5.16 Espandere lo spazio di archiviazione utilizzabile usando collegamenti simbolici

Suggerimento

Questo è un metodo deprecato. Usare invece Sezione 9.5.15, se possibile.

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 dir-da-usare vecchia-dir
$ sudo mkdir -p /percorso/della/dir-vuota
$ sudo ln -sf /percorso/della/dir-vuota dir-da-usare
$ sudo cp -a vecchia-dir/* dir-da-usare
$ sudo rm -rf vecchia-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.



Attenzione

Certo software può non funzionare bene con un "collegamento simbolico ad una directory".

9.5.17 Espandere lo spazio di archiviazione utilizzabile usando overlayfs

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=/percorso/della/vecchia,upperdir=/percorso/della/vuota,workdir=/percorso/di/ lavoro ↵
```

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 Immagine del disco

Questa sezione tratta della manipolazione di immagini di dischi.

9.6.1 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 disco.img
# dd if=/dev/sdb of=disco.img
```

Si può creare l'immagine del disco del [MBR \(master boot record\)](#) dei PC tradizionali (vedere Sezione [9.5.2](#)), 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.

9.6.2 Scrivere direttamente sul disco

Il file immagine del disco, `"disco.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=disco.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
```

9.6.3 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
...bla...bla...bla
# 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
...bla...bla...bla
# umount partizione.img
```

Ogni partizione di un'immagine di disco "disco.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 # verifica la capacita del kernel
max_part:Maximum number of partitions per loop device
max_loop:Maximum number of loop devices
# losetup -a # verifica che nulla stia usando il device loop
# rmmod loop
# modprobe loop max_part=16
```

Ora il device loop può gestire fino a 16 partizioni.

```
# losetup -v -f disco.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
...bla...bla...bla
# 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 disco.img
...
# mkdir -p /mnt/loop0p2
# mount -t ext4 /dev/mapper/loop0p2 /mnt/loop0p2
...
...bla...bla...bla
```

```
# 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.

9.6.4 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 vecchia; mkdir nuova
# mount -t auto -o loop disco.img vecchia
# dd bs=1 count=0 if=/dev/zero of=nuova.img seek=5G
# mount -t auto -o loop nuova.img nuova
# cd vecchia
# cp -a --sparse=always ./ ../nuova/
# cd ..
# umount nuova.img
# umount disco.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 disco.img
Loop device is /dev/loop3
# zerofree /dev/loop3
# cp --sparse=always disco.img nuovo.img
```

9.6.5 Creare un file con immagine di disco vuoto

Si può creare un'immagine di disco vuota "disco.img", che può crescere fino a 5GiB, usando `dd(1)` nel modo seguente.

```
$ dd bs=1 count=0 if=/dev/zero of=disco.img seek=5G
```

È possibile creare un file system `ext4` in questa immagine di disco, "disco.img" usando il [device loop](#) nel modo seguente.

```
# losetup -f -v disco.img
Loop device is /dev/loop1
# mkfs.ext4 /dev/loop1
...bla...bla...bla
# losetup -d /dev/loop1
$ du --apparent-size -h disco.img
5.0G disco.img
$ du -h disco.img
83M disco.img
```

La dimensione del file "disco.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 [9.6.3](#), si può partizionare tale immagine di disco "disco.img" usando `parted(8)` o `fdisk(8)` e si può creare in essa file system usando `mkfs.ext4(8)`, `mkswap(8)`, ecc.

9.6.6 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 ID_volume -o cd.iso directory_sorgente
```

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 ID_volume \  
-b isolinux/isolinux.bin -c isolinux/boot.cat \  
-no-emul-boot -boot-load-size 4 -boot-info-table directory_sorgente
```

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.

9.6.7 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".

9.6.8 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](#)).

9.7 I dati binari

Questa sezione tratta della manipolazione diretta dei dati binari su supporti di archiviazione.

9.7.1 Visualizzare e modificare dati binari

Il metodo di visualizzazione dei dati binari più basilare è l'uso del comando `od -t x1`.

pacchetto	popcon	dimensione	descrizione
coreutils	V:886, I:999	15016	pacchetto base che contiene <code>od(1)</code> per fare il dump di file (HEX, ASCII, OCTAL, ...)
bsdmainutils	V:864, I:998	586	pacchetto di utilità che contiene <code>hd(1)</code> per fare il dump di file (HEX, ASCII, OCTAL, ...)
hexedit	V:1, I:12	62	visualizzatore ed editor binario (HEX, ASCII)
bless	V:0, I:5	973	editor esadecimale completo (GNOME)
okteta	V:1, I:21	1344	editor esadecimale completo (KDE4)
ncurses-hexedit	V:0, I:2	126	visualizzatore ed editor binario (HEX, ASCII, EBCDIC)
beav	V:0, I:1	133	visualizzatore ed editor binario (HEX, ASCII, EBCDIC, OCTAL, ...)

Tabella 9.20: Elenco di pacchetti che visualizzano e modificano dati binari

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 dei [mainframe IBM](#).

9.7.2 Manipolare file senza montare i dischi

Esistono strumenti per leggere e scrivere file senza montare i dischi.

pacchetto	popcon	dimensione	descrizione
mtools	V:13, I:106	361	utilità per file MSDOS senza montarli
hfsutils	V:0, I:8	1842	utilità per file HFS e HFS+ senza montarli

Tabella 9.21: Elenco di pacchetti per manipolare file senza montare i dischi

9.7.3 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:2, I:10	246	insiemi di volumi di archivi di parità, per controllare e riparare file
dvdaster	V:0, I:2	1588	protezione contro perdita di dati/graffi/invecchiamento di supporti CD/DVD
dvbackup	V:0, I:0	412	strumento di backup che usa camcorder MiniDV (fornisce rsbep(1))
vdmfec	V:0, I:0	97	recupera blocchi perduti usando Forward Error Correction

Tabella 9.22: Elenco di strumenti per aggiungere dati ridondanti a file

9.7.4 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:39	1315	utilità per scansione delle partizione e ripristino di dischi
magicrescue	V:0, I:3	220	utilità per ripristinare file cercando byte magici
scalpel	V:0, I:4	82	strumento di escavazione di file frugale e ad alte prestazioni
myrescue	V:0, I:3	38	recupera dati da dischi fissi danneggiati
extundelete	V:1, I:11	148	utilità per decancellare file da file system ext3/4
ext4magic	V:0, I:3	232	utilità per decancellare file da file system ext3/4
ext3grep	V:0, I:3	278	strumento per aiutare a recuperare file cancellati da file system ext3
scrounge-ntfs	V:0, I:3	45	programma di recupero di dati da file system NTFS
gzrt	V:0, I:0	57	insieme di strumenti di recupero di gzip
sleuthkit	V:1, I:14	1113	strumenti per analisi forensi (Sleuthkit)
autopsy	V:0, I:2	1021	interfaccia grafica per SleuthKit
foremost	V:0, I:7	123	applicazione forense per il recupero dei dati
guymager	V:0, I:1	1044	strumento forense per immagini basato su Qt
dcfldd	V:0, I:5	94	versione migliorata di dd per analisi forensi e sicurezza

Tabella 9.23: 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_deleted_inodes` e `unde1` di `debugfs(8)` nel pacchetto `e2fsprogs`.

9.7.5 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 file_grande
$ cat x* >file_grande
```

**Attenzione**

Assicurarsi di non avere altri file che iniziano con "x" per evitare conflitti nei nomi.

9.7.6 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_da_pulire
```

9.7.7 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
$ : > semprezero.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.

9.7.8 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.

9.7.9 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/pippo
# cd /mnt/pippo
# dd if=/dev/zero of=spazzatura
dd: writing to 'spazzatura': 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.

9.7.10 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 pippo > pluto
$ less pluto
$ ps aux | grep 'less[ ]'
bozo    4775  0.0  0.0  92200   884 pts/8    S+   00:18   0:00 less pluto
$ rm pluto
$ ls -l /proc/4775/fd | grep bar
lr-x----- 1 bozo bozo 64 2008-05-09 00:19 4 -> /home/bozo/pluto (deleted)
$ cat /proc/4775/fd/4 >pluto
$ ls -l
-rw-r--r-- 1 bozo bozo 4 2008-05-09 00:25 pluto
$ cat pluto
pippo
```

Quando si ha il pacchetto `lsOf` installato, eseguire in un altro terminale quanto segue.

```
$ ls -li pluto
2228329 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:02 pluto
$ lsOf |grep pluto|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/pluto
$ rm pluto
$ lsOf |grep pluto|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/pluto (deleted)
$ cat /proc/4775/fd/4 >pluto
$ ls -li pluto
2228302 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:05 pluto
$ cat pluto
pippo
```


9.7.11 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 paperino
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 pippo
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 pluto
```

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 /percorso/del/punto/di/mount -xdev -inum 2738404
```

9.7.12 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
```

9.8 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 4.7.4). 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](#) e [eCryptfs](#) facilitano la cifratura automatica dei dati in modo nativo attraverso moduli del kernel Linux con un minimo lavoro da parte dell'utente.

pacchetto	popcon	dimensione	descrizione
cryptsetup	V:39, I:80	360	utilità per device a blocchi cifrati (dm-crypt / LUKS)
cryptmount	V:3, I:4	224	utilità per device a blocchi cifrati (dm-crypt / LUKS) con particolare attenzione al montaggio/smontaggio da parte di utenti normali
ecryptfs-utils	V:6, I:9	396	utilità per file system stack cifrati (eCryptfs)

Tabella 9.24: Elenco di utilità per la cifratura dei dati

[Dm-crypt](#) è un file system crittografico che usa [device-mapper](#). [Device mapper](#) mappa un blocco di un device ad un altro.

[eCryptfs](#) è un altro file system crittografico che usa un file system stack; quest'ultimo si sovrappone ad una directory esistente in un file system montato.



Attenzione

La cifratura dei dati consuma tempo CPU, ecc. Valutare bene i suoi costi e i benefici.

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](#).

9.8.1 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.

```
# badblocks -c 1024 -s -w -t random -v /dev/sdx
# fdisk /dev/sdx
... "n" "p" "1" "return" "return" "w"
# cryptsetup luksFormat /dev/sdx1
...
# cryptsetup open --type luks /dev/sdx1 sdx1
...
# ls -l /dev/mapper/
total 0
crw-rw---- 1 root root 10, 60 2008-10-04 18:44 control
brw-rw---- 1 root disk 254, 0 2008-10-04 23:55 sdx1
# mkfs.vfat /dev/mapper/sdx1
...
# cryptsetup luksClose sdx1
```

Si può poi montarla come una chiavetta qualunque in `"/media/<etichetta_disco>"`, tranne per il fatto che verrà chiesta la password (vedere Sezione [10.1.7](#)) nei moderni ambienti desktop, come GNOME che usa `gnome-mount(1)`. La differenza è che ogni dato scritto in essa è cifrato. In alternativa è possibile formattare il supporto con un file system diverso, ad esempio `ext4` usando `"mkfs.ext4 /dev/sdx1"`.

Nota

Se si è veramente paranoici per ciò che riguarda la sicurezza dei propri dati, potrebbe essere necessario sovrascrivere diverse volte (il comando `"badblocks"` nell'esempio precedente). Questa operazione richiede però parecchio tempo.

9.8.2 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
# /etc/init.d/cryptdisks restart
...
# swapon -a
```

9.8.3 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
```

9.8.4 Cifratura automatica di file con eCryptfs

È possibile cifrare automaticamente i file scritti in `"~/Private/"` usando [eCryptfs](#) e il pacchetto `ecryptfs-utils`.

- Eseguire `ecryptfs-setup-private(1)` e impostare `"~/Private/"` nei prompt mostrati.
- Attivare `"~/Private/"` eseguendo `ecryptfs-mount-private(1)`.
- Spostare i file con dati sensibili in `"~/Private/"` e creare i collegamenti simbolici necessari.
 - File candidati sono: `"~/fetchmailrc"`, `"~/ssh/identity"`, `"~/ssh/id_rsa"`, `"~/ssh/id_dsa"` e altri file con permessi `"go-rwx"`
- Spostare le directory con dati sensibili in una sottodirectory in `"~/Private/"` e creare i collegamenti simbolici necessari.
 - Directory candidate sono: `"~/gnupg"` e altre directory con permessi `"go-rwx"`
- Creare un collegamento simbolico da `"~/Desktop/Private/"` a `"~/Private/"` per facilitare le operazioni dal desktop.
- Disattivare `"~/Private/"` eseguendo `ecryptfs-umount-private(1)`.
- Attivare `"~/Private/"` eseguendo `ecryptfs-mount-private(1)` quando sono necessari dati cifrati.

Suggerimento

Dato che [eCryptfs](#) cifra selettivamente solo i file sensibili, il suo costo in termini di risorse di sistema è molto minore dell'uso di [dm-crypt](#) sull'intero device root o `"/home"`. Non richiede nessun lavoro speciale di archiviazione su disco, ma non può mantenere confidenziali tutti i metadati del file system.

9.8.5 Montare automaticamente eCryptfs

Se si usa la propria password di login come wrapper per le chiavi di cifratura, si può automatizzare il montaggio di eCryptfs tramite [PAM \(Pluggable Authentication Modules\)](#).

Inserire la riga seguente immediatamente prima di `"pam_permit.so"` in `"/etc/pam.d/common-auth"`.

```
auth required pam_ecryptfs.so unwrap
```

Inserire la riga seguente come ultima riga in `"/etc/pam.d/common-session"`.

```
session optional pam_ecryptfs.so unwrap
```

Insiere la riga seguente come prima riga attiva in `/etc/pam.d/common-password`.

```
password required pam_ecryptfs.so
```

Ciò risulta molto comodo.

**avvertimento**

Errori di configurazione di [PAM](#) possono lasciare l'utente chiuso fuori dal proprio sistema. Vedere Capitolo [4](#).

**Attenzione**

Se si usa la propria password di login per fare da wrapper alle chiavi di cifratura, i dati cifrati sono sicuri tanto quanto la password di login dell'utente (vedere Sezione [4.3](#)). A meno che non si scelga prudentemente una [password forte](#), i propri dati saranno a rischio se qualcuno esegue software di [forzatura della password](#) dopo aver rubato il portatile (vedere Sezione [4.7.4](#)).

9.9 Il kernel

Debian distribuisce, per le architetture supportate, kernel Linux modulari contenuti in pacchetti.

9.9.1 Kernel Linux 2.6/3.x

Nel kernel Linux 2.6/3.x ci sono alcune funzionalità degne di nota rispetto alla versione 2.4.

- I device vengono creati dal sistema udev (vedere Sezione [3.3](#)).
- L'accesso in lettura/scrittura ai dispositivi CD/DVD IDE non usa il modulo `ide-scsi`.
- Le funzioni di filtraggio dei pacchetti di rete usando i moduli `iptables` del kernel.

Il salto di versione da Linux 2.6.39 a Linux 3.0 non è dovuto a cambiamenti tecnologici fondamentali, ma al ventesimo anniversario.

9.9.2 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 [9.6.3](#))

Vedere `kernel-parameters.txt(.gz)` e altri documenti correlati nella documentazione del kernel Linux (`/usr/share/doc`) fornita dai pacchetti `linux-doc-3.*`.

9.9.3 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

Per compilare alcuni programmi specifici per il kernel, come moduli per il kernel da fonti esterne e il demone `automounter` (`amd`), è necessario includere nella propria riga di comando il percorso ai corrispondenti header del kernel, ad esempio `"-I/usr/src/linux-particular-version/include/"`. `module-assistant(8)` (o la sua forma abbreviata `m-a`) aiuta l'utente a compilare ed installare pacchetti di moduli in modo semplice per uno o più kernel personalizzati.

9.9.4 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:451	20	pacchetti essenziali per compilare pacchetti Debian: <code>make</code> , <code>gcc</code> , ...
bzip2	V:198, I:942	184	utilità di compressione e decompressione per file <code>bz2</code>
libncurses5-dev	V:15, I:149	991	librerie di sviluppo e documentazione per <code>ncurses</code>
git	V:271, I:438	32161	<code>git</code> : sistema distribuito di controllo delle revisioni usato dal kernel Linux
fakeroot	V:32, I:501	215	fornisce un ambiente <code>fakeroot</code> per compilare pacchetti da utente non root
initramfs-tools	V:319, I:990	105	strumento per compilare un <code>initramfs</code> (specifico di Debian)
dkms	V:105, I:207	282	DKMS (Dynamic Kernel Module Support, supporto dinamico per i moduli del kernel) (generico)
devscripts	V:9, I:65	2148	script di aiuto per i manutentori di pacchetti Debian (specifico di Debian)

Tabella 9.25: 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)`, `mkinitramfs(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`.

Nota

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.

9.9.5 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_<arch>" 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: <http://kernel-handbook.alioth.debian.org>

9.9.6 Driver per hardware e firmware

Un driver hardware è il codice eseguito nel sistema target. La maggior parte dei driver hardware sono ora disponibili come software libero e sono inclusi nei normali pacchetti Debian dei kernel nell'area `main`.

- 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`)

Il firmware è il codice o i dati caricati sul dispositivo (ad esempio [microcodice](#) della CPU, codice di rendering eseguito nella GPU oppure dati [FPGA/CPLD](#), ...). Alcuni pacchetti firmware sono disponibili come software libero, ma molti pacchetti firmware non lo sono dato che contengono dati binari senza sorgenti.

- `firmware-linux-free` (`main`)
- `firmware-linux-nonfree` (`non-free`)
- `firmware-linux-*` (`non-free`)
- `*-firmware` (`non-free`)
- `intel-microcode` (`non-free`)
- `amd64-microcode` (`non-free`)

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](#).

9.10 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>.

9.10.1 Strumenti per la virtualizzazione

Ci sono diversi pacchetti in Debian relativi alla [virtualizzazione](#) ed [emulazione](#) di sistema che vanno oltre il semplice [chroot](#). Alcuni pacchetti aiutano anche nell'impostazione di tali sistemi.

Vedere l'articolo di Wikipedia [Comparison of platform virtual machines](#) per una comparazione dettagliata di diverse soluzioni per la virtualizzazione di piattaforme.

9.10.2 Fasi del processo di virtualizzazione

Nota

Alcune funzionalità descritte in seguito sono disponibili solamente in squeeze o rilasci successivi.

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).
-

pacchetto	popcon	dimensione	descrizione
schroot	V:7, I:10	2659	strumento specializzato per eseguire pacchetti Debian binari in chroot
sbuild	V:1, I:4	283	strumento per compilare pacchetti Debian binari da sorgenti Debian
pbuilder	V:2, I:18	958	creatore di pacchetti personali per pacchetti Debian
debootstrap	V:6, I:66	261	avviare un sistema Debian base (scritto in sh)
cdebootstrap	V:0, I:3	112	avviare un sistema Debian (scritto in C)
virt-manager	V:8, I:33	7722	Virtual Machine Manager : applicazione desktop per gestire macchine virtuali
libvirt-clients	V:28, I:49	1921	programmi per la libreria libvirt
bochs	V:0, I:1	4086	Bochs : emulatore PC IA-32
qemu	I:39	666	QEMU : veloce emulatore generico di processore
qemu-system	I:41	92	QEMU : binari per emulazione completa del sistema
qemu-user	V:3, I:40	61725	QEMU : binari per emulazione in spazio utente
qemu-utils	V:10, I:98	5220	QEMU : utilità
qemu-kvm	V:20, I:70	101	KVM : virtualizzazione completa su hardware x86 con virtualizzazione assistita da hardware
virtualbox	V:34, I:44	75363	VirtualBox : soluzione per virtualizzazione i dx86 su i386 e amd64
xen-tools	V:0, I:6	666	strumenti per gestire server virtuali XEN Debian
wine	V:24, I:102	180	Wine : implementazione della API Windows (suite standard)
dosbox	V:3, I:20	2706	DOSBox : emulatore x86 con grafica Tandy/Herc/CGA/EGA/VGA/SVGA, suono e DOS
dosemu	V:0, I:4	4891	DOSEMU : emulatore DOS di Linux
vzctl	V:2, I:3	1060	OpenVZ , soluzione per virtualizzazione server - strumenti di controllo
vzquota	V:2, I:3	221	OpenVZ , soluzione per virtualizzazione server - strumenti per quote
lxc	V:8, I:14	2076	strumenti in spazio utente per contenitori Linux containers

Tabella 9.26: Elenco di strumenti di virtualizzazione

- 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 9.6.1 e Sezione 9.6.5).
- 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 9.6.3).
 - Per le immagini disco supportate da [QEMU](#), montarle come [device a blocchi di rete](#) (vedere Sezione 9.10.3).
- Popolare il file system obiettivo con i dati di sistema necessari.
 - L'uso di programmi come `debootstrap` e `cdebootstrap` aiuta questo processo (vedere Sezione 9.10.4).
 - 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](#).

9.10.3 Montare il file immagine di disco virtuale

Per i file immagine raw di disco, vedere Sezione 9.6.

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 9.6.3). 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)`.

9.10.4 Sistema chroot

`chroot(8)` offre il metodo più basilare per eseguire diverse istanze dell'ambiente GNU/Linux simultaneamente in un singolo sistema senza riavviare.



Attenzione

Gli esempi seguenti presuppongono che entrambi i sistemi, quello genitore e quello `chroot`, condividano la stessa architettura.

Si può imparare a impostare ed usare chroot(8) eseguendo il programma pbuilder(8) in script(1) nel modo seguente.

```
$ sudo mkdir /sid-root
$ sudo pbuilder --create --no-targz --debug --buildplace /sid-root
```

Si può vedere come debootstrap(8) o cdebootstrap(1) popoli i dati di sistema per l'ambiente sid in `"/sid-root"`.

Suggerimento

L'installatore Debian usa debootstrap(8) o cdebootstrap(1) per l'[installazione Debian](#). Possono anche essere usati per installare Debian in un sistema senza usare un disco di installazione Debian, ma invece uno di un'altra distribuzione GNU/Linux.

```
$ sudo pbuilder --login --no-targz --debug --buildplace /sid-root
```

Si può vedere come venga creata una shell di sistema in esecuzione nell'ambiente sid, nel modo seguente.

1. Copiare la configurazione locale (`"/etc/hosts"`, `"/etc/hostname"`, `"/etc/resolv.conf"`)
2. Montare il file system `"/proc"`
3. Montare il file system `"/dev/pts"`
4. Creare `"/usr/sbin/policy-rc.d"` che restituisca sempre il codice 101
5. Eseguire `"chroot /sid-root bin/bash -c 'exec -a -bash bin/bash'"`

Nota

Alcuni programmi in chroot per funzionare possono aver bisogno dell'accesso ad più file nel sistema genitore di quanti ne fornisca pbuilder. Per esempio, per essere montati con bind o copiati, possono essere necessari `"/sys"`, `"/etc/passwd"`, `"/etc/group"`, `"/var/run/utmp"`, `"/var/log/wtmp"`, ecc.

Nota

Il file `"/usr/sbin/policy-rc.d"` evita che, in un sistema Debian, i programmi demone vengano avviati automaticamente. Vedere `"/usr/share/doc/sysv-rc/README.policy-rc.d.gz"`.

Suggerimento

Lo scopo originale del pacchetto specializzato chroot, pbuilder è di costruire un sistema chroot e di compilare un pacchetto all'interno del chroot. È un sistema ideale per controllare che le dipendenze di compilazione di un pacchetto siano corrette e per assicurare che, nel pacchetto risultante, non ci siano dipendenze sbagliate o superflue.

Suggerimento

Il pacchetto simile schroot può dare un'idea di come eseguire un sistema chroot i386 in un sistema genitore amd64.

9.10.5 Sistemi desktop multipli

Io raccomando l'uso di [QEMU](#) o [VirtualBox](#) per eseguire, in un sistema Debian `stable`, sistemi desktop multipli in maniera sicura usando la [virtualizzazione](#). Permettono di eseguire applicazioni desktop di Debian `unstable` e `testing` senza i comuni rischi ad essi associati.

Dato che [QEMU](#) puro è molto lento, è raccomandata la sua velocizzazione con [KVM](#) quando il sistema host lo permette.

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).

Nota

Gli archivi tar(1) compressi con gzip usano l'estensione di file ".tgz" o ".tar.gz".

Nota

Gli archivi tar(1) compressi con xz usano l'estensione di file ".txz" o ".tar.xz".

Nota

La popolarità dei metodi di compressione negli strumenti FOSS come tar(1) è cambiata nel tempo nel modo seguente gzip → bzip2 → xz

Nota

cp(1), scp(1) e tar(1) possono avere alcune limitazioni per file speciali. cpio(1) è più versatile.

Nota

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.

Nota

La struttura interna dei file di dati di LibreOffice è quella dei file «.jar» che può essere aperta anche da unzip.

Nota

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
-

pacchetto	popcon	dimensione	estensione	comando	commento
tar	V:915, I:999	2770	.tar	tar(1)	strumento di archiviazione standard (standard de facto)
cpio	V:409, I:998	989	.cpio	cpio(1)	strumento di archiviazione Unix in stile System V, da usare con find(1)
binutils	V:191, I:707	89	.ar	ar(1)	strumento di archiviazione per la creazione di librerie statiche
fastjar	V:5, I:53	172	.jar	fastjar(1)	strumento di archiviazione per Java (simile a zip)
pax	V:16, I:45	175	.pax	pax(1)	nuovo strumento POSIX di archiviazione, compromesso tra tar e cpio
gzip	V:884, I:999	231	.gz	gzip(1) , zcat(1) , ...	utilità GNU di compressione LZ77 (standard de facto)
bzip2	V:198, I:942	184	.bz2	bzip2(1) , bzip2(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:4, I:51	141	.lzma	lzma(1)	utilità di compressione LZMA con maggiore rapporto di compressione di gzip(1) (deprecata)
xz-utils	V:432, I:956	515	.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)
p7zip	V:78, I:360	934	.7z	7zr(1) , p7zip(1)	strumento di archiviazione file 7-Zip con alto rapporto di compressione (compressione LZMA)
p7zip-full	V:149, I:523	4407	.7z	7z(1) , 7za(1)	strumento di archiviazione file 7-Zip con alto rapporto di compressione (compressione LZMA e altre)
lzop	V:6, I:48	97	.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:52, I:429	608	.zip	zip(1)	InfoZIP : strumento di archiviazione e compressione per DOS
unzip	V:349, I:800	534	.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:886, I:999	15016	GNU cp	copia file e directory localmente ("a" per modalità ricorsiva)
openssh-client	V:822, I:995	4147	scp	copia file e directory da remoto (client, "r" per modalità ricorsiva)
openssh-server	V:684, I:811	870	sshd	copia file e directory da remoto (server remoto)
rsync	V:222, I:609	691	-	sincronizzazione e backup in remoto unidirezionale
unison	V:4, I:19	3452	-	sincronizzazione e backup in remoto bidirezionale

Tabella 10.2: Elenco di strumenti di copia e sincronizzazione

Suggerimento

L'esecuzione, tramite cron(8), dello script bkup citato in Sezione [10.2.3](#) con l'opzione "-g1", dovrebbe fornire funzionalità molto simili a dumpfs di Plan9 per un archivio di dati statici.

Suggerimento

Gli strumenti di controllo delle versioni VCS (Version control system) in Tabella [10.11](#) 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 archivio.tar.xz ./sorgente
$ tar -xvJf archivio.tar.xz
```

In alternativa usare i comandi seguenti.

```
$ find ./sorgente -xdev -print0 | tar -cvJf archivio.tar.xz --null -F -
```

cpio(1):

```
$ find ./sorgente -xdev -print0 | cpio -ov --null > archivio.cpio; xz archivio.cpio
$ zcat archivio.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 ./sorgente; rsync -aHAXSv . /dest
# cd ./sorgente; rsync -aHAXSv . utente@host.dom:/dest
```

In alternativa si può usare la sintassi con `"una barra / alla fine della directory sorgente"`.

```
# rsync -aHAXSv ./sorgente/ /dest
# rsync -aHAXSv ./sorgente/ utente@host.dom:/dest
```

In alternativa usare i comandi seguenti.

```
# cd ./sorgente; find . -print0 | rsync -aHAXSv0 --files-from=- . /dest
# cd ./sorgente; find . -print0 | rsync -aHAXSv0 --files-from=- . utente@host.dom:/dest
```

GNU cp(1) e openSSH scp(1):

```
# cd ./sorgente; cp -a . /dest
# cd ./sorgente; scp -pr . utente@host.dom:/dest
```

GNU tar(1):

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

cpio(1):

```
# cd ./sorgente; 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 9.3.9). 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 [casaforti 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`.

- Il pacchetto `udisks` fornisce un demone e le utilità associate per montare e smontare questi dispositivi.
- [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 `"uhelp="` 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`.

Il punto di mount negli ambienti desktop moderni viene scelto come `/media/<etichetta_disco>` che può essere personalizzata nel modo seguente.

- `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 8.4.6).

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.

file system	descrizione dello scenario di uso tipico
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)
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)
file system MINIX	archiviazione, efficiente in termini di spazio, di file dati unix su dischetti floppy
file system ext2	condivisione di dati su dispositivi come dischi fissi con sistemi Linux più vecchi
file system ext3	condivisione di dati su dispositivi come dischi fissi con sistemi Linux più vecchi
file system ext4	condivisione di dati su dispositivi come dischi fissi con sistemi Linux recenti

Tabella 10.3: Elenco di possibili scelte per il file system di dispositivi di archiviazione removibili con scenari di uso tipici

Suggerimento

Vedere Sezione 9.8.1 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 [9.5.2](#)) 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 [6.10](#) e Sezione [6.11](#).

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 smb.conf(5) 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 exports(5) 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 9.7 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 6.3).

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 9.3.14 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.
- Backup regolari dei dati degli utenti possono essere realizzati da un semplice script (Sezione 10.2.2) e `cron(8)`.
- [Bacula](#), [Amanda](#) e [BackupPC](#) sono suite di utilità di backup complete che sono pensate per backup regolari in rete.

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:7	341	dump(8) e restore(8) di BSD 4.4 per file system ext2/ext3/ext4
xfsdump	V:0, I:11	838	dump e ripristino con xfsdump(8) e xfsrestore(8) per file system XFS in GNU/Linux e IRIX
backupninja	V:4, I:4	329	sistema di meta-backup leggero ed estensibile
bacula-common	V:10, I:19	2055	Bacula : backup, ripristino e controllo in rete - file comuni di supporto
bacula-client	I:5	164	Bacula : backup, ripristino e controllo in rete - metapacchetto client
bacula-console	V:1, I:7	65	Bacula : backup, ripristino e controllo in rete - console testuale
bacula-server	I:2	164	Bacula : backup, ripristino e controllo in rete - metapacchetto server
amanda-common	V:1, I:2	9964	Amanda : Advanced Maryland Automatic Network Disk Archiver (Librerie)
amanda-client	V:1, I:2	1072	Amanda : Advanced Maryland Automatic Network Disk Archiver (Client)
amanda-server	V:0, I:0	1044	Amanda : Advanced Maryland Automatic Network Disk Archiver (Server)
backup-manager	V:1, I:2	543	strumento di backup a riga di comando
backup2l	V:1, I:1	113	strumento di backup/ripristino per supporti montabile (basato su dischi) che richiede bassa manutenzione
backupper	V:4, I:5	2232	BackupPC è un sistema ad alte prestazioni di qualità professionale per il backup di PC (basato su dischi)
duplicity	V:8, I:16	1538	backup incrementali (remoti)
flexbackup	V:0, I:0	242	backup incrementali (remoti)
rdiff-backup	V:9, I:18	704	backup incrementali (remoti)
restic	V:0, I:0	16046	backup incrementali (remoti)
rsnapshot	V:6, I:12	452	backup incrementali (remoti)
slbackup	V:0, I:0	146	backup incrementali (remoti)

Tabella 10.5: Elenco di suite con utilità di backup

10.2.2 Uno script di esempio per il backup di sistema

Per un sistema desktop Debian personale con la suite `unstable`, devo proteggere solo i dati personali e quelli critici; tanto reinstallo comunque il sistema una volta all'anno. Perciò non ho ragione di fare il backup dell'intero sistema o di installare un'utilità completa di backup.

Uso un semplice script per fare un archivio di backup e masterizzarlo su CD/DVD usando una GUI. Ecco uno script di esempio che fa proprio questo.

```
#!/bin/sh -e
# Copyright (C) 2007-2008 Osamu Aoki <osamu@debian.org>, Public Domain
BUUID=1000; USER=osamu # UID and name of a user who accesses backup files
BUDIR="/var/backups"
XDIR0=".+/Mail|.+/Desktop"
XDIR1=".+/.thumbnails|.+/.?Trash|.+/.?[cC]ache|.+/.gvfs|.+/.sessions"
XDIR2=".+/CVS|.+/.git|.+/.svn|.+/.Downloads|.+/.Archive|.+/.Checkout|.+/.tmp"
XSFX=".+\.iso|.+\.tgz|.+\.tar\.gz|.+\.tar\.bz2|.+\.cpio|.+\.tmp|.+\.swp|.+~"
SIZE="+99M"
DATE=$(date --utc +"%Y%m%d-%H%M")
[ -d "$BUDIR" ] || mkdir -p "$BUDIR"
umask 077
dpkg --get-selections \* > /var/lib/dpkg/dpkg-selections.list
debconf-get-selections > /var/cache/debconf/debconf-selections

{
find /etc /usr/local /opt /var/lib/dpkg/dpkg-selections.list \
    /var/cache/debconf/debconf-selections -xdev -print0
find /home/$USER /root -xdev -regextype posix-extended \
    -type d -regex "$XDIR0|$XDIR1" -prune -o -type f -regex "$XSFX" -prune -o \
    -type f -size "$SIZE" -prune -o -print0
find /home/$USER/Mail/Inbox /home/$USER/Mail/Outbox -print0
find /home/$USER/Desktop -xdev -regextype posix-extended \
    -type d -regex "$XDIR2" -prune -o -type f -regex "$XSFX" -prune -o \
    -type f -size "$SIZE" -prune -o -print0
} | cpio -ov --null -O $BUDIR/BU$DATE.cpio
chown $BUUID $BUDIR/BU$DATE.cpio
touch $BUDIR/backup.stamp
```

Questo è pensato come esempio di script che deve essere eseguito da root.

Modificarlo ed eseguirlo nel modo seguente.

- Modificare lo script per includere tutti i propri dati importanti (vedere Sezione [10.1.5](#) e Sezione [10.2](#)).
- Sostituire `"find ...-print0"` con `"find ...-newer $BUDIR/backup.stamp -print0"` per fare backup incrementali.
- Trasferire i file di backup su un host remoto usando `scp(1)` o `rsync(1)` o masterizzarli su CD/DVD per una maggiore sicurezza dei dati. (Io uso la GUI del desktop GNOME per masterizzare i CD/DVD.. Vedere Sezione [12.1.8](#) per l'inserimento di dati di ridondanza.)

Mantenere le cose semplici!

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"`.

10.2.3 Uno script di copia per backup di dati

Per l'insieme dei dati in un albero di directory, la copia con "cp -a" fornisce normali backup.

Per un grande insieme di dati statici non sovrascritti in un albero di directory, come quello nella directory "/var/cache/apt/packages" i collegamenti fisici creati con "cp -al" forniscono un'alternativa ai normali backup con un uso efficiente dello spazio su disco.

Ecco uno script di copia, che ho chiamato bkup, per il backup di dati. Questo script copia tutti i file (non-VCS) contenuti nella directory attuale in una directory datata nella directory genitore o in un host remoto.

```
#!/bin/sh -e
# Copyright (C) 2007-2008 Osamu Aoki <osamu@debian.org>, Public Domain
fdot(){ find . -type d \( -iname ".?*" -o -iname "CVS" \) -prune -o -print0; }
fall(){ find . -print0; }
mkdircd(){ mkdir -p "$1"; chmod 700 "$1"; cd "$1">/dev/null; }
FIND="fdot"; OPT="-a"; MODE="CPIO"; HOST="localhost"; EXTP="$(hostname -f)"
BKUP="$(basename $(pwd)).bkup"; TIME="$(date +%Y%m%d-%H%M%S)"; BU="$BKUP/$TIME"
while getopts gcCsStrllaXe:h:T f; do case $f in
g) MODE="GNUCP";; # cp (GNU)
c) MODE="CPIO";; # cpio -p
C) MODE="CPIOI";; # cpio -i
s) MODE="CPIOSSH";; # cpio/ssh
t) MODE="TARSSH";; # tar/ssh
r) MODE="RSYNCSH";; # rsync/ssh
l) OPT="-alv";; # hardlink (GNU cp)
L) OPT="-av";; # copy (GNU cp)
a) FIND="fall";; # find all
A) FIND="fdot";; # find non CVS/ .???/
x) set -x;; # trace
e) EXTP="{OPTARG}";; # hostname -f
h) HOST="{OPTARG}";; # user@remotehost.example.com
T) MODE="TEST";; # test find mode
\?) echo "use -x for trace."
esac; done
shift $(expr $OPTIND - 1)
if [ $# -gt 0 ]; then
  for x in $@; do cp $OPT $x $x.$TIME; done
elif [ $MODE = GNUCP ]; then
  mkdir -p "../$BU"; chmod 700 "../$BU"; cp $OPT . "../$BU/"
elif [ $MODE = CPIO ]; then
  mkdir -p "../$BU"; chmod 700 "../$BU"
  $FIND|cpio --null --sparse -pvd ../$BU
elif [ $MODE = CPIOI ]; then
  $FIND|cpio -ov --null | ( mkdircd "../$BU"&&cpio -i )
elif [ $MODE = CPIOSSH ]; then
  $FIND|cpio -ov --null|ssh -C $HOST "( mkdircd \"$EXTP/$BU\"&&cpio -i )"
elif [ $MODE = TARSSH ]; then
  (tar cvf - . )|ssh -C $HOST "( mkdircd \"$EXTP/$BU\"&&tar xvpf - )"
elif [ $MODE = RSYNCSH ]; then
  rsync -aHAXsv ./ "${HOST}:${EXTP}-${BKUP}-${TIME}"
else
  echo "Any other idea to backup?"
  $FIND |xargs -0 -n 1 echo
fi
```

Questo script è pensato per essere un esempio per i comandi; leggerlo e modificarlo prima di usarlo.

Suggerimento

Io tengo questo script bkup nella mia directory "/usr/local/bin/" ed eseguo il comando bkup senza opzioni nella directory di lavoro ogni volta che ho bisogno di un backup istantaneo temporaneo.

Suggerimento

Per creare uno storico delle istantanee di un albero di file sorgenti o di un albero di file di configurazione, l'uso di `git(7)` è più semplice e più efficiente in termini di spazio (vedere Sezione [10.6.5](#)).

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.

pacchetto	popcon	dimensione	comando	descrizione
gnupg	V:698, I:998	693	<code>gpg(1)</code>	GNU Privacy Guard - strumento OpenPGP di cifratura e firma
gpgv	V:881, I:999	807	<code>gpgv(1)</code>	GNU Privacy Guard - strumento di verifica delle firme
paperkey	V:0, I:2	55	<code>paperkey(1)</code>	estrae solamente le informazioni segrete da chiavi OpenPGP segrete
cryptsetup	V:39, I:80	360	<code>cryptsetup(8)</code> , ...	utilità per la cifratura dm-crypto per i device a blocchi con supporto LUKS
ecryptfs-utils	V:6, I:9	396	<code>ecryptfs(7)</code> , ...	utilità per cifratura di file system stack ecryptfs
coreutils	V:886, I:999	15016	<code>md5sum(1)</code>	calcola e controlla message digest MD5
coreutils	V:886, I:999	15016	<code>sha1sum(1)</code>	calcola e controlla message digest SHA1
openssl	V:794, I:990	1303	<code>openssl(1ssl)</code>	calcola message digest con "openssl dgst" (OpenSSL)

Tabella 10.6: Elenco di strumenti per l'infrastruttura di sicurezza dei dati

Per [dm-crypto](#) e [ecryptfs](#) che implementano infrastrutture di cifratura automatica dei dati attraverso moduli del kernel Linux, vedere Sezione [9.8](#).

10.3.1 Gestione delle chiavi per GnuPG

Quelli che seguono sono alcuni comandi per [GNU Privacy Guard](#) per la gestione base delle chiavi.

comando	descrizione
<code>gpg --gen-key</code>	Genera una nuova chiave
<code>gpg --gen-revoke mio_ID_utente</code>	genera una chiave di revoca per mio_ID_utente
<code>gpg --edit-key user_ID</code>	modifica la chiave in modo interattivo, "help" per l'aiuto
<code>gpg -o file --export</code>	esporta tutte le chiavi in un file
<code>gpg --import file</code>	importa tutte le chiavi da un file
<code>gpg --send-keys ID_utente</code>	invia la chiave di ID_utente al server di chiavi
<code>gpg --recv-keys ID_utente</code>	riceve la chiave di ID_utente dal server di chiavi
<code>gpg --list-keys ID_utente</code>	elenca le chiavi di ID_utente
<code>gpg --list-sigs ID_utente</code>	elenca le firme di ID_utente
<code>gpg --check-sigs ID_utente</code>	controlla le firme di ID_utente
<code>gpg --fingerprint ID_utente</code>	controlla le impronte digitali di ID_utente
<code>gpg --refresh-keys</code>	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.

Il comando seguente carica la mia chiave "1DD8D791" sul popolare server di chiavi "hkp://keys.gnupg.net".

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

```
$ 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

Il plugin `gnupg` permette di eseguire GnuPG in modo trasparente per i file con estensione `.gpg`, `.asc` e `.pgp`.

```
# aptitude install vim-scripts vim-addon-manager
$ vim-addons install gnupg
```

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.

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 pippo@example.org
<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

```
$ md5sum pippo pluto >paperino.md5
$ cat paperino.md5
d3b07384d113edec49eaa6238ad5ff00  pippo
c157a79031e1c40f85931829bc5fc552  pluto
$ md5sum -c paperino.md5
pippo: OK
pluto: 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.vecchio file.nuovo > file.patch0
$ diff -u vecchio/file nuovo/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 Aggiornare con fusioni a 3 vie

Se si hanno tre versioni di un codice sorgente, è possibile effettuare una fusione a 3 vie in maniera efficace usando diff3(1) nel modo seguente.

```
$ diff3 -m file.mio file.vecchio file.tuo > file
```

10.5 Sistemi di controllo delle versioni

Ecco un riassunto dei [sistemi di controllo delle versioni \(VCS\)](#) nel sistema Debian.

Nota

Se non si ha esperienza di sistemi VCS, si dovrebbe iniziare ad imparare con **Git** che sta acquistando sempre più popolarità.

pacchetto	popcon	dimensione	comando	descrizione
diffutils	V:866, I:982	1394	diff(1)	confronta i file riga per riga
diffutils	V:866, I:982	1394	diff3(1)	confronta e fonde tre file riga per riga
vim	V:117, I:396	2507	vimdiff(1)	confronta 2 file uno di fianco all'altro in vim
patch	V:132, I:837	216	patch(1)	applica un file diff ad un originale
dpatch	V:1, I:15	191	dpatch(1)	gestisce serie di patch per pacchetti Debian
diffstat	V:19, I:187	70	diffstat(1)	produce un istogramma delle modifiche apportate da un diff
patchutils	V:19, I:180	223	combinediff(1)	crea una patch cumulativa da due patch incrementali
patchutils	V:19, I:180	223	dehtmldiff(1)	estrae un diff da una pagina HTML
patchutils	V:19, I:180	223	filterdiff(1)	estrae o esclude diff da un file diff
patchutils	V:19, I:180	223	fixcvsdiff(1)	aggiusta file diff creati da CVS che sono male interpretati da patch(1)
patchutils	V:19, I:180	223	flipdiff(1)	scambia l'ordine di due patch
patchutils	V:19, I:180	223	grepdiff(1)	mostra quali file siano modificati da una patch che fa corrispondenza con un'espressione regolare
patchutils	V:19, I:180	223	interdiff(1)	mostra le differenze tra due file diff unificati
patchutils	V:19, I:180	223	lsdiff(1)	mostra quali file vengano modificati da una patch
patchutils	V:19, I:180	223	recountdiff(1)	ricalcola conteggi e offset in diff unificati
patchutils	V:19, I:180	223	rediff(1)	aggiusta conteggi ed offset di un diff modificato a mano
patchutils	V:19, I:180	223	splitdiff(1)	separa due patch incrementali
patchutils	V:19, I:180	223	unwrapdiff(1)	ripristina patch il cui contenuto è stato mandato a capo automaticamente
wiggle	V:0, I:0	166	wiggle(1)	applica le patch respinte
quilt	V:3, I:43	711	quilt(1)	gestisce serie di patch
meld	V:15, I:41	3116	meld(1)	confronta e fonde file (GTK)
dirdiff	V:0, I:2	144	dirdiff(1)	mostra le differenze ed apporta i cambiamenti tra alberi di directory
docdiff	V:0, I:0	573	docdiff(1)	confronta due file parola per parola / carattere per carattere
imediff2	V:0, I:0	34	imediff2(1)	strumento interattivo a tutto schermo di applicazione di modifiche bidirezionale
makepatch	V:0, I:0	102	makepatch(1)	genera file patch estesi
makepatch	V:0, I:0	102	applypatch(1)	applica file patch estesi
wdiff	V:5, I:81	643	wdiff(1)	mostra le differenze di parole tra file di testo

Tabella 10.10: Elenco di strumenti per la fusione di codice sorgente

pacchetto	popcon	dimensione	strumento	tipo di VCS	commento
cssc	V:0, I:2	1979	CSSC	locale	clone di SCCS Unix (deprecato)
rcs	V:3, I:23	539	RCS	locale	" SCCS Unix fatto bene"
cvs	V:6, I:57	4597	CVS	remoto	precedente standard per VCS remoto
subversion	V:30, I:150	4678	Subversion	remoto	"CVS fatto bene", il nuovo standard di fatto per VCS remoto
git	V:271, I:438	32161	Git	distribuito	veloce DVCS in C (usato dal kernel Linux ed altri)
mercurial	V:11, I:62	548	Mercurial	distribuito	DVCS in Python e un po' di C
bazaar	V:4, I:23	73	Bazaar	distribuito	DVCS influenzato da tla , scritto in Python (usato da Ubuntu)
darcs	V:0, I:8	31349	Darcs	distribuito	DVCS con algebra intelligente per le patch (lento)
tla	V:0, I:7	1011	GNU arch	distribuito	DVCS principalmente di Tom Lord (storico)
monotone	V:0, I:0	5815	Monotone	distribuito	DVCS in C++
tkcvs	V:0, I:1	1498	CVS, ...	remoto	visualizzazione GUI di alberi di archivi VCS (CVS, Subversion, RCS)
gitk	V:7, I:50	1474	Git	distribuito	visualizzazione GUI di alberi di repository VCS (Git)

Tabella 10.11: Elenco di strumenti per sistemi di controllo delle versioni

Un VCS viene a volte chiamato sistema di controllo delle revisioni (RCS) o gestione della configurazione software (SCM).

Al giorno d'oggi i VCS distribuiti come Git sono gli strumenti preferiti. CVS e Subversion possono essere ancora utili per aggregarsi ad alcune realtà di programmi open source esistenti.

Debian fornisce servizi VCS liberi attraverso il [servizio Debian Alioth](#) che supporta praticamente tutti i VCS. La sua documentazione è reperibile su <http://wiki.debian.org/Alioth>.

Ci sono alcune nozioni base da ricordare per creare un accesso condiviso ad un archivio VCS.

- Usare "umask 002" (vedere Sezione [1.2.4](#))
- Far sì che tutti i file dell'archivio VCS appartengano ad un gruppo pertinente
- Abilitare l'impostazione dell'ID di gruppo per tutte le directory dell'archivio VCS (schema di creazione di file in stile BSD, vedere Sezione [1.2.3](#))
- Far sì che gli utenti che condividano l'archivio VCS appartengano al gruppo

10.5.1 Comparazione di comandi VCS

Quella che segue è una comparazione ipersemplificata, per dare un'idea generale dei comandi VCS nativi. La sequenza di comandi usata tipicamente può richiedere opzioni ed argomenti.



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

Se c'è un file eseguibile git-pippo nel percorso specificato da \$PATH, l'inserimento nella riga di comando di "git pippo" senza trattino invoca tale git-pippo. Questa è una funzionalità del comando git.

Git	CVS	Subversion	funzione
git init	cvs init	svn create	creazione dell'archivio (locale)
-	cvs login	-	login all'archivio remoto
git clone	cvs co	svn co	fare il check out dell'archivio remoto come albero di directory di lavoro
git pull	cvs up	svn up	aggiornare l'albero di lavoro fondendovi l'archivio remoto
git add .	cvs add	svn add	aggiungere il/i file nell'albero di lavoro al VCS
git rm	cvs rm	svn rm	rimuovere il/i file nell'albero di lavoro dal VCS
-	cvs ci	svn ci	fare il commit dei cambiamenti all'archivio remoto
git commit -a	-	-	fare il commit dei cambiamenti all'archivio locale
git push	-	-	aggiornare l'archivio remoto con l'archivio locale
git status	cvs status	svn status	visualizza lo stato dell'albero di lavoro dal VCS
git diff	cvs diff	svn diff	diff <archivio_diriferimento> <albero_di_lavoro>
git repack -a -d; git prune	-	-	reimpacchetta l'archivio locale in un singolo pacchetto
gitk	tkcvs	tkcvs	visualizzazione GUI dell'albero dell'archivio VCS

Tabella 10.12: Comparazione di comandi VCS nativi

Suggerimento

Gli strumenti con interfaccia utente grafica, come tkcvs(1) e gitk(1), aiutano realmente a tenere traccia della cronologia delle revisioni dei file. L'interfaccia web fornita da molti archivi pubblici per la navigazione dei loro repository è anch'essa piuttosto utile.

Suggerimento

Git può lavorare direttamente con diversi archivi VCS, come quelli forniti da CVS e Subversion, e fornisce archivi locali per modifiche locali tramite i pacchetti git-cvs e git-svn. Vedere [Git per utenti CVS](#) e la Sezione [10.6.4](#).

Suggerimento

Git ha comandi che non hanno equivalenti in CVS e Subversion: "fetch", "rebase", "cherry-pick", ...

10.6 Git

Git può fare tutto ciò che riguarda la gestione di codice sorgente, sia in locale sia in remoto. Ciò significa che è possibile registrare i cambiamenti al codice sorgente senza bisogno di connessione di rete all'archivio remoto.

10.6.1 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 "Nome Cognome"
$ git config --global user.email proprionome@esempio.com
```

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
```

10.6.2 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)
- [Uso quotidiano di GIT con 20 comandi o poco più](/usr/share/doc/git-doc/everyday.html) (/usr/share/doc/git-doc/everyday.html)
- [git per gli utenti CVS](/usr/share/doc/git-doc/gitcvcs-migration.html) (/usr/share/doc/git-doc/gitcvcs-migration.html)
 - Questo descrive anche come impostare server come CVS ed estrarre vecchi dati da CVS portandoli in Git.
- [Altre risorse git disponibili in rete](#)
 - [Corso veloce per Git - SVN](#)
 - [Git Magic](/usr/share/doc/gitmagic/html/index.html) (/usr/share/doc/gitmagic/html/index.html)

I comandi git-gui(1) e gitk(1) rendono l'uso di Git molto semplice.



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.

10.6.3 Comandi Git

Anche se l'archivio a monte usa un diverso VCS, potrebbe essere una buona idea usare git(1) per l'attività locale, dato che si può gestire la propria copia locale dell'albero dei sorgenti senza una connessione di rete con l'archivio a monte. Ecco alcuni pacchetti e comandi usati con git(1).

Suggerimento

Con git(1) si lavora su un ramo locale con molti commit e si usa successivamente un comando simile a "git rebase -i master" per riorganizzare la cronologia dei cambiamenti. Ciò permette di avere cronologie dei cambiamenti pulite. Vedere git-rebase(1) e git-cherry-pick(1).

Suggerimento

Quando si vuole tornare ad una directory di lavoro pulita senza perdere lo stato attuale della directory di lavoro, si può usare "git stash". Vedere git-stash(1).

pacchetto	popcon	dimensione	comando	descrizione
git-doc	I:19	10603	N/D	documentazione ufficiale per Git
gitmagic	I:1	719	N/D	"Git Magic", una guida per Git più semplice da capire
git	V:271, I:438	32161	git(7)	Git, il sistema di controllo delle revisioni veloce, scalabile e distribuito
gitk	V:7, I:50	1474	gitk(1)	browser degli archivi Git con interfaccia utente grafica e cronologia
git-gui	V:2, I:29	2201	git-gui(1)	interfaccia utente grafica per Git (senza cronologia)
git-svn	V:2, I:28	986	git-svnimport(1)	importa i dati da Subversion in Git
git-svn	V:2, I:28	986	git-svn(1)	fornisce operazioni bidirezionali tra Subversion e Git
git-cvs	V:0, I:13	1107	git-cvsignport(1)	importa i dati da CVS in Git
git-cvs	V:0, I:13	1107	git-cvsexportcommit(1)	esporta un commit a un checkout CVS da Git
git-cvs	V:0, I:13	1107	git-cvsserver(1)	emulatore di server CVS per Git
git-email	V:0, I:13	791	git-send-email(1)	invia una raccolta di patch come messaggio di posta da Git
stgit	V:0, I:1	1535	stg(1)	quilt sopra a git (Python)
git-buildpackage	V:1, I:12	3850	git-buildpackage(1)	automatizza la creazione di pacchetti Debian con Git
guilt	V:0, I:0	146	guilt(7)	quilt sopra a git (SH/AWK/SED/...)

Tabella 10.13: Elenco di pacchetti e comandi relativi a Git

10.6.4 Git per repository Subversion

Si può fare il check out di un repository Subversion su "svn+ssh://svn.example.org/project/module/trunk" in un repository Git locale in ". /dest" fare poi nuovamente il commit sul repository Subversion. Ad esempio:

```
$ git svn clone -s -rHEAD svn+ssh://svn.esempio.org/progetto dest
$ cd dest
... fare i cambiamenti
$ git commit -a
... continuare a lavorare localmente con git
$ git svn dcommit
```

Suggerimento

L'uso di "-rHEAD" permette di evitare di dover clonare tutti i contenuti della cronologia del repository Subversion.

10.6.5 Git per registrare la cronologia della configurazione

Si può registrare manualmente la cronologia della configurazione usando strumenti [Git](#). Quello che segue è un semplice esempio che insegna a registrare il contenuto di "/etc/apt/".

```
$ cd /etc/apt/
$ sudo git init
$ sudo chmod 700 .git
$ sudo git add .
$ sudo git commit -a
```

Fare il commit della configurazione con una descrizione.

Modificare i file di configurazione

```
$ cd /etc/apt/  
$ sudo git commit -a
```

Fare il commit della configurazione con una descrizione e andare avanti con la propria vita.

```
$ cd /etc/apt/  
$ sudo gitk --all
```

Si avrà una completa cronologia della configurazione.

Nota

Per lavorare con tutti i tipi di permessi dei dati di configurazione è necessario `sudo(8)`. Per i dati di configurazione dell'utente, si può saltare l'uso di `sudo`.

Nota

Il comando `"chmod 700 .git"` nell'esempio precedente è necessario per proteggere i dati dell'archivio da accessi non autorizzati.

Suggerimento

Per l'impostazione di un sistema di registrazione della configurazione più completo, guardare il pacchetto `etckeeper`: Sezione [9.2.10](#).

10.7 CVS

Vedere la documentazione seguente.

- `cvs(1)`
- `"/usr/share/doc/cvs/html-cvsclient"`
- `"/usr/share/doc/cvs/html-info"`
- `"/usr/share/doc/cvsbook"`
- `"info cvs"`

10.7.1 Configurazione di archivi CVS

La configurazione seguente permette solo ai membri del gruppo `"src"` di fare commit nell'archivio CVS e solo ai membri del gruppo `"staff"` di amministrare il CVS, riducendo perciò le possibilità di autodanneggiarsi.

```
# cd /var/lib; umask 002; mkdir cvs  
# export CVSR00T=/srv/cvs/progetto  
# cd $CVSR00T  
# chown root:src .  
# chmod 2775 .  
# cvs -d $CVSR00T init  
# cd CVSR00T  
# chown -R root:staff .  
# chmod 2775 .  
# touch val-tags  
# chmod 664 history val-tags  
# chown root:src history val-tags
```

Suggerimento

Si può limitare la creazione di nuovi progetti modificando il proprietario della directory "\$CVSR00T" in "root : staff" e i suoi permessi in "3775".

10.7.2 Accesso locale al CVS

L'archivio CVS predefinito è indicato da "\$CVSR00T". Il comando seguente imposta "\$CVSR00T" per l'accesso locale.

```
$ export CVSROOT=/srv/cvs/progetto
```

10.7.3 Accesso remoto a CVS con pserver

Molti server CVS pubblici forniscono accesso remoto solo in lettura all'account "anonymous" tramite il servizio pserver. I contenuti del sito Debian, per esempio, sono mantenuti dal [progetto webwml](#) attraverso CVS sul servizio Debian Alioth. Il comando seguente imposta "\$CVSR00T" per l'accesso remoto al suo archivio CVS.

```
$ export CVSROOT=:pserver:anonymous@anonscm.debian.org:/cvs/webwml
$ cvs login
```

Nota

Dato che pserver è esposto ad attacchi di spionaggio ed è insicuro, l'accesso in scrittura è solitamente disabilitato dagli amministratori del server.

10.7.4 Accesso remoto a CVS con ssh

Il comando seguente imposta "\$CVS_RSH" e "\$CVSR00T" per l'accesso remoto con SSH all'archivio CVS del [progetto webwml](#).

```
$ export CVS_RSH=ssh
$ export CVSROOT=:ext:account@cvs.alioth.debian.org:/cvs/webwml
```

Si può anche usare l'autenticazione con chiave pubblica per SSH che elimina la richiesta di inserimento della password remota.

10.7.5 Importare un nuovo sorgente in CVS

Creare un nuovo albero di sorgenti locale in "~/percorso/di/modulo1" con il comando seguente.

```
$ mkdir -p ~/percorso/di/modulo1; cd ~/percorso/di/modulo1
```

Popolare il nuovo albero di sorgenti locale in "~/percorso/di/modulo1" con i file.

Importarlo nel CVS con i parametri seguenti.

- Nome modulo: "modulo1"
- Etichetta fornitore: "Ramo-principale" (etichetta per l'intero ramo)
- Etichetta rilascio: "Rilascio-iniziale" (etichetta per un rilascio specifico)

```
$ cd ~/percorso/di/modulo1
$ cvs import -m "Inizio di modulo1" Ramo-principale Rilascio-iniziale
$ rm -Rf . # opzionale
```

10.7.6 Permessi dei file negli archivi CVS

CVS non sovrascrive i file attuali nell'archivio, ma li sostituisce con altri. Perciò i permessi di scrittura della directory dell'archivio sono fondamentali. Per ogni nuovo modulo di `modulo1` nell'archivio in `"/srv/cvs/progetto"`, eseguire i comandi seguenti per assicurare che i permessi siano corretti, se necessario.

```
# cd /srv/cvs/progetto
# chown -R root:src modulo1
# chmod -R ug+rwX modulo1
# chmod 2775 modulo1
```

10.7.7 Sequenza di azioni tipiche nell'uso di CVS

Ecco un esempio di una sequenza tipica di azioni svolte nell'uso di CVS.

Controllare tutti i moduli disponibili dal progetto CVS a cui punta `"$CVSR00T"` con il comando seguente.

```
$ cvs rls
CVSR00T
modulo1
modulo2
...
```

Fare il checkout di `"modulo1"` nella sua directory predefinita `"/modulo1"` con il comando seguente.

```
$ cd ~/percorso/di
$ cvs co modulo1
$ cd modulo1
```

Fare i cambiamenti desiderati necessari ai contenuti.

Controllare i cambiamenti facendo l'equivalente di `"diff -u [archivio] [locale]"` con il comando seguente.

```
$ cvs diff -u
```

Si scopre, ad esempio, di aver corrotto gravemente un file `"file_da_ripristinare"`, ma gli altri file sono a posto.

Sovrascrivere `"file_da_ripristinare"` con una copia pulita dal CVS con il comando seguente.

```
$ cvs up -C file_da_ripristinare
```

Salvare l'albero dei sorgenti locale aggiornato sul CVS con il comando seguente.

```
$ cvs ci -m "Descrivere qui i cambiamenti"
```

Creare ed aggiungere il file `"file_da_aggiungere"` al CVS con il comando seguente.

```
$ vi file_da_aggiungere
$ cvs add file_da_aggiungere
$ cvs ci -m "Aggiunto file_da_aggiungere"
```

Fondere la versione più recente dal CVS con il comando seguente.

```
$ cvs up -d
```

Prestare attenzione alle righe che iniziano con `"C nomefile"` che indicano le modifiche che creano conflitti.

Cercare il codice non modificato in `".#nomefile.versione"`.

Per trovare le modifiche che creano conflitti cercare `"<<<<<<"` e `">>>>>>"` nei file.

Modificare i file in modo da risolvere i conflitti.

Aggiungere un'etichetta di rilascio `"Rilascio-1"` nel modo seguente.

```
$ cvs ci -m "ultimo commit per Rilascio-1"
$ cvs tag Rilascio-1
```

Modificare ulteriormente.

Rimuovere l'etichetta di rilascio "Rilascio-1" con il comando seguente.

```
$ cvs tag -d Rilascio-1
```

Fare il commit delle modifiche sul CVS con il comando seguente.

```
$ cvs ci -m "ultimissimo commit per Rilascio-1"
```

Riaggiungere l'etichetta di rilascio "Rilascio-1" all'HEAD CVS aggiornato principale con il comando seguente.

```
$ cvs tag Rilascio-1
```

Creare un ramo con un'etichetta permanente "Rilascio-iniziale-risoluzionebug" dalla versione originale a cui punta l'etichetta "Rilascio-iniziale" e farne il checkout nella directory "~/percorso/di/vecchio" nel modo seguente.

```
$ cvs rtag -b -r Rilascio-iniziale Rilascio-iniziale-risoluzionebug modulo1
$ cd ~/percorso/di
$ cvs co -r Rilascio-iniziale-risoluzionebug -d vecchio modulo1
$ cd vecchio
```

Suggerimento

Per specificare una particolare data come punto di ramificazione, usare "-D 2005-12-20" (formato [ISO 8601](#) per le date) invece di "-r Rilascio-iniziale".

Lavorare in questo albero locale di sorgenti che ha l'etichetta permanente "Rilascio-iniziale-risoluzionebug" e che è basato sulla versione originale.

Lavorare in questo ramo da soli ... fino a che qualcun altro non si aggiunge al ramo "Rilascio-iniziale-risoluzionebug".

Fare la sincronizzazione dei file modificati da altri in questo ramo, creando contemporaneamente nuove directory se necessario, con il comando seguente.

```
$ cvs up -d
```

Modificare i file in modo da risolvere i conflitti.

Fare il commit delle modifiche sul CVS con il comando seguente.

```
$ cvs ci -m "fatto il commit in questo ramo"
```

Aggiornare l'albero locale con l'HEAD del principale, rimuovendo allo stesso tempo l'etichetta permanente ("-A") e senza espansione delle parole chiave ("-kk"), con il comando seguente.

```
$ cvs up -d -kk -A
```

Aggiornare l'albero locale (contenuto = HEAD del principale) fondendo dal ramo "Rilascio-iniziale-risoluzionebug" e senza espansione delle parole chiave con il comando seguente.

```
$ cvs up -d -kk -j Rilascio-iniziale-risoluzionebug
```

Correggere i conflitti con l'editor.

Fare il commit delle modifiche sul CVS con il comando seguente.

```
$ cvs ci -m "fusione con Rilascio-iniziale-risoluzionebug"
```

Creare un file archivio nel modo seguente.

```
$ cd ..
$ mv vecchio vecchio-modulo1-risoluzionebug
$ tar -cvzf vecchio-modulo1-risoluzionebug.tar.gz vecchio-modulo1-risoluzionebug
$ rm -rf vecchio-modulo1-risoluzionebug
```

Suggerimento

Il comando "cvs up" accetta l'opzione "-d" per creare nuove directory e l'opzione "-P" per eliminare le directory vuote.

Suggerimento

Si può fare il checkout solamente di una sottodirectory di "modulo1" fornendo il suo nome come in "cvs co modulo1/sottodir".

opzione	significato
-n	esecuzione di prova, nessun effetto
-t	visualizza i messaggi che mostrano i passi dell'attività di cvs

Tabella 10.14: Opzioni importanti per comandi CVS (da usare come primi argomenti di cvs(1))

10.7.8 File più recenti dal CVS

Per ottenere i file più recenti dal CVS usare "tomorrow" nel modo seguente.

```
$ cvs ex -D tomorrow nome_modulo
```

10.7.9 Amministrazione del CVS

Aggiungere l'alias "mx" per un modulo ad un progetto CVS (server locale) usando il comando seguente.

```
$ export CVSR00T=/srv/cvs/progetto
$ cvs co CVSR00T/moduli
$ cd CVSR00T
$ echo "mx -a modulo1" >>moduli
$ cvs ci -m "Ora mx e' un alias per modulo1"
$ cvs release -d .
```

Ora si può fare il checkout di "modulo1" (alias: "mx") dal CVS alla directory "nuova" nel modo seguente.

```
$ cvs co -d nuova mx
$ cd nuova
```

Nota

Per seguire la procedura precedente sono necessari gli adeguati permessi sui file.

10.7.10 Bit di esecuzione per il checkout CVS

Quando si fa il checkout di file dal CVS viene mantenuto il loro bit del permesso di esecuzione.

Ogni volta si incontrano problemi con i permessi di esecuzione di un file di cui si è fatto il checkout, ad esempio "nomefile", per risolverli cambiare i suoi permessi nell'archivio CVS corrispondente nel modo seguente.

```
# chmod ugo-x nomefile
```

10.8 Subversion

Subversion è un sistema di controllo delle versioni di **recente generazione** che rimpiazza il più vecchio CVS. Ha la maggior parte delle funzionalità di CVS, tranne le etichette ed i rami.

Per impostare un server Subversion è necessario installare i pacchetti `subversion`, `libapache2-svn` e `subversion-tools`.

10.8.1 Configurazione di archivi Subversion

Attualmente il pacchetto `subversion` non imposta un archivio, perciò è necessario farlo manualmente. Una possibile collocazione per un archivio è in `/srv/svn/progetto`.

Creare una directory con il comando seguente

```
# mkdir -p /srv/svn/progetto
```

Creare il database dell'archivio con il comando seguente.

```
# svnadmin create /srv/svn/progetto
```

10.8.2 Accedere a Subversion tramite il server Apache2

Se si accede all'archivio Subversion solamente attraverso il server Apache2, è necessario soltanto rendere l'archivio scrivibile dal server WWW, nel modo seguente.

```
# chown -R www-data:www-data /srv/svn/progetto
```

Per permettere l'accesso all'archivio tramite autenticazione dell'utente aggiungere (o decommentare) quanto segue in `/etc/apache2/`

```
<Location /progetto>
  DAV svn
  SVNPath /srv/svn/progetto
  AuthType Basic
  AuthName "Subversion repository"
  AuthUserFile /etc/subversion/passwd
<LimitExcept GET PROPFIND OPTIONS REPORT>
  Require valid-user
</LimitExcept>
</Location>
```

Creare un file di autenticazione dell'utente con il comando seguente.

```
# htpasswd2 -c /etc/subversion/passwd un-qualche-nomeutente
```

Riavviare Apache2.

Il nuovo archivio Subversion è accessibile da `svn(1)` agli URL `"http://localhost/progetto"` e `"http://esempio.com/pr"` (assumendo che l'URL del proprio server web sia `"http://esempio.com/"`).

10.8.3 Accesso locale a Subversion da parte di un gruppo

I comandi seguenti impostano l'archivio Subversion per l'accesso locale da parte di un gruppo, ad esempio **progetto**.

```
# chmod 2775 /srv/svn/progetto
# chown -R root:src /srv/svn/progetto
# chmod -R ug+rwX /srv/svn/progetto
```

Il nuovo archivio Subversion è accessibile con `svn(1)` per gli utenti locali che appartengono al gruppo **progetto**, all'URL `"file:///localhost/srv/svn/progetto"` o `"file:///srv/svn/progetto"`. Per assicurare l'accesso al gruppo è necessario usare i programmi come `svn`, `svnserve`, `svnlook` e `svnadmin` con `"umask 002"`.

10.8.4 Accesso remoto a Subversion con SSH

Se esiste un archivio Subversion accessibile da un gruppo all'URL `"esempio.com:/srv/svn/progetto"`, si può accedere ad esso con `svn(1)` e SSH all'URL `"svn+ssh://esempio.com:/srv/svn/progetto"`.

10.8.5 Struttura delle directory di Subversion

Per compensare la mancanza di rami ed etichette in Subversion, molti progetti usano con esso un albero di directory simile al seguente.

```
----- modulo1
| |-- branches
| |-- tags
| | |-- rilascio-1.0
| | '-- rilascio-2.0
| |
| | '-- trunk
| |   |-- file1
| |   |-- file2
| |   '-- file3
|
|-- modulo2
```

Suggerimento

Per marcare i rami e le etichette è necessario usare il comando `"svn copy ..."`. Ciò assicura che Subversion registri la cronologia dei cambiamenti dei file in maniera corretta e risparmi spazio.

10.8.6 Importare un nuovo sorgente in Subversion

Creare un nuovo albero di sorgenti locale in `"~/percorso/di/modulo1"` con il comando seguente.

```
$ mkdir -p ~/percorso/di/modulo1; cd ~/percorso/di/modulo1
```

Popolare il nuovo albero di sorgenti locale in `"~/percorso/di/modulo1"` con i file.

Importarlo In Subversion con i parametri seguenti.

- Nome modulo: `"modulo1"`
- URL sito Subversion: `"file:///srv/svn/progetto"`
- Directory Subversion: `"modulo1/trunk"`

- Etichetta Subversion: "modulo1/tags/Rilascio-iniziale"

```
$ cd ~/percorso/di/modulo1
$ svn import file:///srv/svn/progetto/modulo1/trunk -m "Inizio di modulo1"
$ svn cp file:///srv/svn/progetto/modulo1/trunk file:///srv/svn/progetto/modulo1/tags/ ↵
  Rilascio-iniziale
```

In alternativa usare i comandi seguenti.

```
$ svn import ~/percorso/di/modulo1 file:///srv/svn/progetto/modulo1/trunk -m "Inizio di ↵
  modulo1"
$ svn cp file:///srv/svn/progetto/modulo1/trunk file:///srv/svn/progetto/modulo1/tags/ ↵
  Rilascio-iniziale
```

Suggerimento

Si possono sostituire gli URL come "file:///..." con URL in qualsiasi altro formato, come "http://..." e "svn+ssh://...".

10.8.7 Sequenza di azioni tipiche nell'uso di Subversion

Ecco un esempio di una sequenza tipica di azioni svolte nell'uso di Subversion con il suo client nativo.

Suggerimento

I comandi client forniti dal pacchetto `git-svn` possono offrire un metodo alternativo di lavoro con Subversion con l'uso del comando `git`. Vedere Sezione [10.6.4](#).

Controllare tutti i moduli disponibili dal progetto Subversion a cui punta l'URL "file:///srv/svn/progetto" con il comando seguente.

```
$ svn list file:///srv/svn/progetto
modulo1
modulo2
...
```

Fare il checkout di "modulo1/trunk" in una directory ". / modulo1" con il comando seguente.

```
$ cd ~/percorso/di
$ svn co file:///srv/svn/progetto/modulo1/trunk modulo1
$ cd modulo1
```

Fare i cambiamenti desiderati necessari ai contenuti.

Controllare i cambiamenti facendo l'equivalente di "diff -u [archivio] [locale]" con il comando seguente.

```
$ svn diff
```

Si scopre, ad esempio, di aver corrotto gravemente un file "file_da_ripristinare", ma gli altri file sono a posto.

Sovrascrivere "file_da_ripristinare" con una copia pulita da Subversion con il comando seguente.

```
$ svn revert file_da_ripristinare
```

Salvare l'albero dei sorgenti locale aggiornato su Subversion con il comando seguente.

```
$ svn ci -m "Descrivere qui i cambiamenti"
```

Creare ed aggiungere il file "file_da_aggiungere" a Subversion con il comando seguente.

```
$ vi file_da_aggiungere
$ svn add file_da_aggiungere
$ svn ci -m "Aggiunto file_da_aggiungere"
```

Fondere la versione più recente da Subversion con il comando seguente.

```
$ svn up
```

Prestare attenzione alle righe che iniziano con "C nomefile" che indicano le modifiche che creano conflitti.

Cercare codice non modificato in, ad esempio "nomefile.r6", "nomefile.r9" e "nomefile.mio".

Per trovare le modifiche che creano conflitti cercare "<<<<<<" e ">>>>>>" nei file.

Modificare i file in modo da risolvere i conflitti.

Aggiungere un'etichetta di rilascio "Rilascio-1" nel modo seguente.

```
$ svn ci -m "ultimo commit per Rilascio-1"
$ svn cp file:///srv/svn/progetto/modulo1/trunk file:///srv/svn/progetto/modulo1/tags/ ↵
  Rilascio-1
```

Modificare ulteriormente.

Rimuovere l'etichetta di rilascio "Rilascio-1" con il comando seguente.

```
$ svn rm file:///srv/svn/progetto/modulo1/tags/Rilascio-1
```

Fare il commit delle modifiche su Subversion con il comando seguente.

```
$ svn ci -m "ultimissimo commit per Rilascio-1"
```

Riaggiungere l'etichetta di rilascio "Rilascio-1" all'HEAD CVS aggiornato di trunk con il comando seguente.

```
$ svn cp file:///srv/svn/progetto/modulo1/trunk file:///srv/svn/progetto/modulo1/tags/ ↵
  Rilascio-1
```

Creare un ramo con percorso "modulo1/branches/Rilascio-iniziale-risoluzionebug" dalla versione originale a cui punta il percorso "modulo1/tags/Rilascio-iniziale" e farne il checkout nella directory "~/percorso/di/vecchio" nel modo seguente.

```
$ svn cp file:///srv/svn/progetto/modulo1/tags/Rilascio-iniziale file:///srv/svn/progetto/ ↵
  modulo1/branches/Rilascio-iniziale-risoluzionebug
$ cd ~/percorso/di
$ svn co file:///srv/svn/progetto/modulo1/branches/Rilascio-iniziale-risoluzionebug vecchio
$ cd vecchio
```

Suggerimento

Per specificare una particolare data come punto di ramificazione, usare "modulo1/trunk@{2005-12-20}" (formato [ISO 8601](#) per le date) invece di "modulo1/tags/Rilascio-iniziale".

Lavorare in questo albero locale di sorgenti che punta al ramo "Rilascio-iniziale-risoluzionebug" e che è basato sulla versione originale.

Lavorare in questo ramo da soli ... fino a che qualcun altro non si aggiunge al ramo "Rilascio-iniziale-risoluzionebug".

Fare la sincronizzazione con i file in questo ramo modificati da altri, con il comando seguente.

```
$ svn up
```

Modificare i file in modo da risolvere i conflitti.

Fare il commit delle modifiche su Subversion con il comando seguente.

```
$ svn ci -m "fatto il commit in questo ramo"
```

Aggiornare l'albero locale con HEAD di trunk nel modo seguente.

```
$ svn switch file:///srv/svn/progetto/modulo1/trunk
```

Aggiornare l'albero locale (contenuto = HEAD di trunk) fondendo dal ramo "Rilascio-iniziale-risoluzionebug" con il comando seguente.

```
$ svn merge file:///srv/svn/progetto/modulo1/branches/Rilascio-iniziale-risoluzionebug
```

Correggere i conflitti con l'editor.

Fare il commit delle modifiche su Subversion con il comando seguente.

```
$ svn ci -m "fusione con Rilascio-iniziale-risoluzionebug"
```

Creare un file archivio nel modo seguente.

```
$ cd ..  
$ mv vecchio vecchio-modulo1-risoluzionebug  
$ tar -cvzf vecchio-modulo1-risoluzionebug.tar.gz vecchio-modulo1-risoluzionebug  
$ rm -rf vecchio-modulo1-risoluzionebug
```

Suggerimento

Si possono sostituire gli URL come "file:///..." con URL in qualsiasi altro formato, come "http://..." e "svn+ssh://...".

Suggerimento

Si può fare il checkout solamente di una sottodirectory di "modulo1" fornendo il suo nome come in "svn co file:///srv/svn/progetto/modulo1/trunk/sottodir modulo1/sottodir", ecc.

opzione	significato
- -dry-run	esecuzione di prova, nessun effetto
-v	visualizza messaggi dettagliati sull'attività di svn

Tabella 10.15: Opzioni importanti per comandi Subversion (da usare come primi argomenti di svn(1))

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:939, I:999	11065	set di caratteri	convertitore della codifica del testo tra localizzazioni usando iconv(1) (fondamentale)
recode	V:4, I:31	608	set caratteri+carattere fine riga	convertitore della codifica del testo tra localizzazioni (versatile, maggior numero di alias e funzionalità)
konwert	V:2, I:59	123	set di caratteri	convertitore della codifica tra localizzazioni (elegante)
nkf	V:0, I:12	347	set di caratteri	traduttore del set di caratteri per il giapponese
tcs	V:0, I:0	479	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:2, I:32	51	carattere fine riga	convertitore del formato di testo tra DOS e Unix: fromdos(1) e todos(1)
macutils	V:0, I:1	298	carattere fine riga	convertitore del formato di testo tra Macintosh e Unix: frommac(1) e tomac(1)

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 codifica1 -t codifica2 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

Alcune codifiche non sono gestite in fase di conversione dei dati e non sono utilizzate come valori di localizzazione (Sezione [8.4.1](#)).

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 "trovato non-UTF-8"
```

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 8.4.6), 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	<code>^J</code>	10	0A
MSDOS e Windows	CR-LF	<code>^M^J</code>	13 10	0D 0A
Macintosh di Apple	CR	<code>^M</code>	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	<code>bsdmainutils</code>	<code>coreutils</code>
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.sito-remoto.com/help-info.html >filetesto
```

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:184, I:605	2301	html → testo puro	conversione da HTML a testo semplice con il comando "w3m -dump"
html2text	V:21, I:62	257	html → testo puro	convertitore avanzato da HTML a testo semplice (ISO 8859-1)
lynx	V:30, I:107	1903	html → testo puro	conversione da HTML a testo semplice con il comando "lynx -dump"
elinks	V:17, I:31	1583	html → testo puro	conversione da HTML a testo semplice con il comando "elinks -dump"
links	V:20, I:45	2123	html → testo puro	conversione da HTML a testo semplice con il comando "links -dump"
links2	V:2, I:17	5366	html → testo puro	conversione da HTML a testo semplice con il comando "links2 -dump"
antiword	V:7, I:13	623	MSWord → testo puro, ps	converte file MSWord in testo puro o ps
catdoc	V:62, I:93	675	MSWord → testo puro, TeX	converte file MSWord in testo puro o TeX
pstotext	V:4, I:6	126	ps/pdf → testo puro	estrae testo da file PostScript e PDF
unhtml	V:0, I:0	42	html → testo puro	rimuove i tag marcatori da file HTML
odt2txt	V:4, I:7	53	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:20, I:437	28345	evidenziare	MACRO Vim per convertire codice sorgente in HTML con <code>":source \$VIMRUNTIME/syntax/html.vim"</code>
cxref	V:0, I:0	1174	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:1, I:7	2008	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:16	1028	evidenziare	converte molti codici sorgenti in file HTML, XHTML, RTF, LaTeX, TeX o XSL-FO con evidenziazione (C++)
grc	V:0, I:2	106	testo → colori	colorazione generica per tutto (Python)
txt2html	V:0, I:4	296	testo → html	convertitore da testo semplice in HTML (Perl)
markdown	V:0, I:7	57	testo → html	formattatore markdown di documenti di testo in (X)HTML (Perl)
asciidoc	I:14	80	testo → tutto	formattatore AsciiDoc di documenti di testo in XML/HTML (Python)
pandoc	V:4, I:34	50901	testo → tutto	convertitore universale di marcatori (Haskell)
python-docutils	V:40, I:360	1752	testo → tutto	formattatore di documenti ReStructured Text in XML (Python)
txt2tags	V:0, I:1	951	testo → tutto	conversione di documenti da testo semplice a HTML, SGML, LaTeX, pagine man, MoinMoin, Magic Point e PageMaker (Python)
udo	V:0, I:0	564	testo → tutto	utilità universale di elaborazione di documenti - testo (linguaggio C)
stx2any	V:0, I:0	264	testo → tutto	convertitore di documenti da testo semplice strutturato in altri formati (m4)
rest2web	V:0, I:0	527	testo → html	convertitore di documenti da ReStructured Text in HTML (Python)
aft	V:0, I:0	235	testo → tutto	sistema di preparazione di documenti a "modello libero" (Perl)
yodl	V:0, I:0	615	testo → tutto	pre-linguaggio per documenti e strumenti per elaborarlo (linguaggio C)
sdf	V:0, I:0	1445	testo → tutto	semplice analizzatore di documenti (Perl)
sisu	V:0, I:0	5341	testo → tutto	infrastruttura per strutturazione, pubblicazione e ricerca di documenti (Ruby)

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.

```
<nome attributo="valore">contenuto</nome>
```

Gli elementi XML senza contenuto sono indicati nella seguente forma breve.

```
<nome attributo="valore"/>
```

La parte `"attributo="valore"` negli esempi precedenti è opzionale.

In XML una sezione di commento è marcata nel modo seguente.

```
<!-- commento -->
```

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

Quando vengono usate entità definite in stile SGML, ad esempio `&qualche-tag;`, la prima definizione prevale su tutte le altre. La definizione di entità è espressa nella forma `<!ENTITY qualche-tag "valore entità">`.

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.

pacchetto	popcon	dimensione	parola chiave	descrizione
docbook-xml	I:521	2131	xml	DTD (Document Type Definition) XML per DocBook
xsltproc	V:16, I:116	149	xslt	elaboratore a riga di comando XSLT (XML → XML, HTML, testo semplice, ecc.)
docbook-xsl	V:14, I:224	14998	xml/xslt	fogli di stile XSL per elaborare XML DocBook con XSLT in vari formati di output
xmlto	V:2, I:33	126	xml/xslt	convertitore da-XML-a-tutto con XSLT
dbtoepub	V:0, I:1	71	xml/xslt	convertitore da XML DocBook a .epub
dblatex	V:6, I:22	4648	xml/xslt	converte file DocBook con XSLT in documenti DVI, PostScript, PDF
fop	V:3, I:47	64	xml/xsl-fo	converte file XML DocBook in PDF

Tabella 11.8: Elenco di strumenti XML

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
openjade	V:3, I:44	988	dsssl	elaboratore standard DSSSL ISO/IEC 10179:1996 (più recente)
docbook-dsssl	V:2, I:34	2604	xml/dsssl	fogli di stile DSSSL per elaborare XML DocBook con DSSSL in vari formati di output
docbook-utils	V:1, I:23	281	xml/dsssl	utilità per file DocBook, inclusa la conversione in altri formati (HTML, RTE, 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 `ye1p` 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.

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.

pacchetto	popcon	dimensione	parola chiave	descrizione
wv	V:6, I:8	717	MSWord → tutto	convertitore di documenti da Microsoft Word a HTML, LaTeX, ecc.
texi2html	V:0, I:10	1832	texi → html	convertitore da texinfo a HTML
man2html	V:0, I:3	141	pagine man → html	convertitore da pagine man a HTML (supporto CGI)
unrtf	V:2, I:4	145	rtf → html	convertitore di documenti da RTF a HTML, ecc.
info2www	V:2, I:4	156	info → html	convertitore da GNU info a HTML (supporto CGI)
ooo2dbk	V:0, I:1	217	sxw → xml	convertitore da documenti SXW OpenOffice.org in XML DocBook
wp2x	V:0, I:0	215	WordPerfect → tutto	da file WordPerfect 5.0 e 5.1 a TeX, LaTeX, troff, GML e HTML
doclifter	V:0, I:0	457	troff → xml	convertitore da troff a XML DocBook

Tabella 11.10: Elenco di strumenti di estrazione di dati XML

pacchetto	popcon	dimensione	parola chiave	descrizione
libxml2-utils	V:22, I:311	167	xml ↔ html ↔ xhtml	strumento XML a riga di comando con xmllint(1) (controllo di sintassi, riformattazione, eliminazione sporcizia, ...)
tidy	V:2, I:17	83	xml ↔ html ↔ xhtml	controllore della sintassi e riformattatore per HTML

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:6, I:65	69	(La)TeX	sistema TeX per impaginazione, anteprima e stampa
groff	V:6, I:125	9362	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](#).

- [teTeX 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 una_qualche_pagina | 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:1, I:17	184	SGML → pagina man	convertitore da SGML DocDook in macro roff man
help2man	V:1, I:10	461	testo → pagina man	generatore automatico di pagine man da --help
info2man	V:0, I:0	134	info → pagina man	convertitore da GNU info a POD o pagine man
txt2man	V:0, I:1	65	testo → pagina man	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.

La licenza dell'ultima versione a monte di Ghostscript di Artifex è passata da AFPL a GPL e sono state fuse in un rilascio unificato tutte le più recenti modifiche alle versioni ESP, come quelle relative a CUPS nel rilascio 8.60.

pacchetto	popcon	dimensione	descrizione
ghostscript	V:159, I:677	219	interprete PostScript/PDF GPL Ghostscript
ghostscript-x	V:31, I:74	211	interprete PostScript/PDF GPL Ghostscript - supporto per display X
libpoppler72	V:18, I:48	3268	libreria di rendering per PDF risultato di un fork del visualizzatore di PDF xpdf
libpoppler-glib8	V:208, I:522	441	libreria di rendering per PDF (libreria condivisa basata su GLib)
poppler-data	V:101, I:667	12219	CMaps per la libreria di rendering per PDF (per supporto CJK : Adobe-*)

Tabella 11.14: Elenco di interpreti PostScript Ghostscript

Suggerimento

La configurazione di Ghostscript può essere visualizzata con "gs -h".

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 pippo1.ps pippo2.ps
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -sOutputFile=bla.pdf -f pippo1.pdf pippo2.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.

pacchetto	popcon	dimensione	parola chiave	descrizione
poppler-utils	V:46, I:479	578	pdf → ps, testo, ...	utilità PDF: pdftops , pdfinfo , pdfimages , pdftotext , pdf fonts
psutils	V:9, I:173	219	ps → ps	strumenti di conversione di documenti PostScript
poster	V:0, I:7	49	ps → ps	crea grandi poster da pagine PostScript
enscript	V:2, I:25	2111	testo → ps, html, rtf	converte testo ASCII in PostScript, HTML, RTF o Pretty-Print
a2ps	V:2, I:24	3620	testo → ps	convertitore e creatore di belle stampe "Da tutto a PostScript"
pdftk	V:9, I:57	3873	pdf → pdf	strumento di conversione di documenti PDF: pdftk
html2ps	V:0, I:5	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:7	478	latex → rtf	converte documenti da LaTeX a RTF leggibili da MS Word
ps2eps	V:6, I:107	94	ps → eps	converte da PostScript a EPS (Encapsulated PostScript)
e2ps	V:0, I:0	112	testo → ps	convertitore da testo a PostScript con supporto per la codifica giapponese
impose+	V:0, I:1	180	ps → ps	Utilità PostScript
trueprint	V:0, I:0	138	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	26	ps → svg	convertitore da PDF al formato SVG (Scalable Vector Graphics)
pdftoipe	V:0, I:0	61	ps → ipe	convertitore da PDF al formato XML di IPE

Tabella 11.15: Elenco di utilità per dati stampabili

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 nomefile
```

```
$ lpr -#3 -o Collate=True nomefile
```

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
sharutils	V:7, I:96	1385	posta	<code>shar(1)</code> , <code>unshar(1)</code> , <code>uuencode(1)</code> , <code>uudecode(1)</code>
mpack	V:2, I:24	91	MIME	codifica e decodifica di messaggi MIME : <code>mpack(1)</code> e <code>munpack(1)</code>
tnef	V:7, I:11	98	ms-tnef	spacchettamento di allegati MIME di tipo "application/ms-tnef" che è un formato esclusivo di Microsoft
uudeview	V:0, I:5	97	posta	codificatore e decodificatore per i seguenti formati: uuencode , xxencode , BASE64 , quoted printable e BinHex
readpst	I:1	21	PST	converte file PST di Outlook di Microsoft nel formato mbox

Tabella 11.16: Elenco di pacchetto che aiutano a convertire dati di posta

Suggerimento

Se il software del programma di posta può essere configurato per usarlo, anche il server [IMAP4](#) (Internet Message Access Protocol, versione 4) (vedere Sezione [6.7](#)) può essere usato per spostare la posta da sistemi proprietari.

11.5.1 Nozioni di base sui dati di posta

I dati di posta ([SMTP](#)) dovrebbero essere limitati a serie di 7 bit di dati. Perciò dati binari e testi a 8 bit sono codificati in un formato a 7 bit con [MIME \(Multipurpose Internet Mail Extensions\)](#) e la selezione del set di caratteri (vedere Sezione [8.4.1](#)).

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

Se i dati di posta non-Unix sono accessibili da un software di posta non-Debian che può comunicare con il server IMAP4, potrebbe essere possibile spostarli eseguendo un proprio server IMAP4 (vedere Sezione [6.7](#)).

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 mpack(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 image-magick(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.

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.

Si possono estrarre i dati dal formato RPM anche nel modo seguente.

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

pacchetto	popcon	dimensione	parola chiave	descrizione
gimp	V:61, I:497	16421	immagine(bitmap)	GNU Image Manipulation Program, programma GNU di manipolazione di immagini
imagemagick	V:77, I:551	196	immagine(bitmap)	programmi di manipolazione immagini
graphicsmagick	V:8, I:16	4966	immagine(bitmap)	programmi di manipolazione di immagini (fork di Imagemagick)
xsane	V:21, I:192	909	immagine(bitmap)	frontend X11 basato su GTK+ per SANE (Scanner Access Now Easy)
netpbm	V:29, I:555	4302	immagine(bitmap)	strumenti di conversione di dati grafici
icoutils	V:11, I:136	194	png ↔ ico(bitmap)	converte icone e puntatori MS Windows da e verso il formato PNG (favicon.ico)
scribus	V:6, I:27	19061	ps/pdf/SVG/...	editor DTP Scribus
libreoffice-draw	V:337, I:475	9762	immagine(vettoriale)	LibreOffice per ufficio LibreOffice - disegno
inkscape	V:165, I:340	129802	immagine(vettoriale)	editor SVG (Scalable Vector Graphics)
dia	V:25, I:38	3910	immagine(vettoriale)	diagram editor (Gtk)
xfig	V:12, I:18	1791	immagine(vettoriale)	strumento per la generazione interattiva di figure in X11
pstoedit	V:8, I:226	666	ps/pdf → immagine(vettoriale)	convertitore di file PostScript e PDF in grafica vettoriale modificabile (SVG)
libwmf-bin	V:13, I:346	104	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:3, I:17	453	immagine → immagine	strumento di post-elaborazione per pagine scansionate per OCR
tesseract-ocr	V:4, I:30	1012	immagine → testo	software OCR libero basato sul motore OCR commerciale di HP
tesseract-ocr-eng	V:1, I:31	37486 4030	immagine → testo	dati per motore OCR: file di lingua tesseract-ocr per testi in inglese
gocr	V:1, I:21	477	immagine → testo	software OCR libero
ocrad	V:1, I:6	299	immagine → testo	software OCR libero
eog	V:90, I:315	11328	immagine(Exif)	visualizzatore di immagini grafiche Eye of GNOME
gthumb	V:15, I:25	3359	immagine(Exif)	navigatore e visualizzatore di immagini (GNOME)
geeqie	V:15, I:24	1989	immagine(Exif)	visualizzatore d'immagini che usa GTK+
shotwell	V:20, I:194	6635	immagine(Exif)	organizzatore di foto digitali (GNOME)
gtkam	V:0, I:7	1154	immagine(Exif)	applicazione per recuperare dati multimediali da fotocamere digitali (GTK+)
gphoto2	V:1, I:14	967	immagine(Exif)	client gphoto2 a riga di comando per le fotocamere digitali
gwenview	V:31, I:104	10871	immagine(Exif)	visualizzatore di immagini (KDE)
kamera	I:104	748	immagine(Exif)	supporto per fotocamere digitali in applicazioni KDE
digikam	V:3, I:16	1832	immagine(Exif)	applicazione per la gestione delle foto digitali per KDE
exiv2	V:5, I:64	246	immagine(Exif)	strumento di manipolazione di metadati EXIF/IPTC
exiftran	V:2, I:25	71	immagine(Exif)	trasforma immagini JPEG di fotocamere digitali
jhead	V:1, I:12	105	immagine(Exif)	manipola la parte non-immagine di file JPEG (fotografie digitali) aderenti conformi ad Exif
exif	V:1, I:11	236	immagine(Exif)	utilità a riga di comando per mostrare informazioni EXIF in file JPEG
exiftags	V:0, I:4	288	immagine(Exif)	utilità per leggere i tag EXIF da un file JPEG di una fotocamera digitale
exifprobe	V:0, I:3	490	immagine(Exif)	legge metadati da immagini digitali
dcraw	V:3, I:24	384	immagine(Raw)	depedifica immagini raw di fotocamere digitali
findimagedupes	V:0, I:1	79	immagine → immagine	cerca immagini simili visivamente o duplicati
ale	V:0, I:0	753	immagine → immagine	fonde immagini per migliorarne la fedeltà o creare mosaici
imageindex	V:0, I:0	144	immagine(Exif)	genera gallerie HTML statiche da immagini
outguess	V:0, I:2	216	jpeg,png	strumento steganografico universale
librecad	V:12, I:18	8039	DXF	editor di dati CAD (KDE)
blender	V:4, I:32	106599	blend, TIFF, VRML, ...	editor di contenuti 3D per animazioni, ecc.
			mc3d, obi	

pacchetto	popcon	dimensione	parola chiave	descrizione
alien	V:4, I:55	166	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:38	86119	tutto → EPUB	convertitore e gestione di una biblioteca per libri elettronici

Tabella 11.18: Elenco di strumenti di conversione di dati vari

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.

pacchetto	popcon	dimensione	documentazione
autoconf	V:34, I:255	1868	"info autoconf" fornito da autoconf-doc
automake	V:33, I:250	1710	"info automake" fornito da automake1.10-doc
bash	V:848, I:999	5798	"info bash" fornito da bash-doc
bison	V:10, I:110	2061	"info bison" fornito da bison-doc
cpp	V:402, I:800	42	"info cpp" fornito da cpp-doc
ddd	V:0, I:12	3965	"info ddd" fornito da ddd-doc
exuberant-ctags	V:7, I:42	333	exuberant-ctags(1)
flex	V:9, I:99	1174	"info flex" fornito da flex-doc
gawk	V:402, I:506	2199	"info gawk" fornito da gawk-doc
gcc	V:189, I:605	45	"info gcc" fornito da gcc-doc
gdb	V:19, I:133	7928	"info gdb" fornito da gdb-doc
gettext	V:66, I:361	6496	"info gettext" fornito da gettext-doc
gfortran	V:7, I:65	16	"info gfortran" fornito da gfortran-doc (Fortran 95)
fpc	I:4	119 119	fpc(1) e html forniti da fp-docs (Pascal)
glade	V:1, I:10	2261	aiuto fornito attraverso menu (compilatore UI)
libc6	V:939, I:999	11065	"info libc" fornito da glibc-doc e glibc-doc-reference
make	V:185, I:612	1211	"info make" fornito da make-doc
xutils-dev	V:1, I:16	1466	imake(1) , xmkmf(1) , ecc.
mawk	V:357, I:997	183	mawk(1)
perl	V:555, I:995	568	perl(1) e pagine html forniti da perl-doc e perl-doc-html
python	V:617, I:987	624	python(1) e pagine html forniti da python-doc
tcl	V:29, I:434	21	tcl(3) e pagine dettagliate del manuale forniti da tcl-doc
tk	V:30, I:422	21	tk(3) e pagine dettagliate di manuale forniti da tk-doc
ruby	V:149, I:340	38	ruby(1) e guida interattiva di riferimento forniti da ri
vim	V:117, I:396	2507	menu di aiuto(F1) fornito da vim-doc
susv2	I:0	15	scarica le specifiche " The Single UNIX Specifications v2 "
susv3	I:0	15	scarica le specifiche " The Single UNIX Specifications v3 "

Tabella 12.1: Elenco di pacchetti di aiuto per la programmazione

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 in-

stallato 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.

**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
... righe di comando
```

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 funzionano 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

Molti script di sistema possono essere interpretati da una qualsiasi delle shell [POSIX](#) (vedere Tabella 1.13). La shell predefinita per il sistema è "/bin/sh" che è un collegamento simbolico che punta al programma reale.

- bash(1) per lenny o precedenti
- dash(1) per squeeze o successivi

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)`.

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.
-

Buono: POSIX	Da evitare: bashismo
if ["\$pippo" = "\$pluto"] ; then ...	if ["\$pippo" == "\$pluto"] ; then ...
diff -u file.c.orig file.c	diff -u file.c{.orig,}
mkdir /pippopluto /pippopaperino	mkdir /pippo{pluto,paperino}
nomefunzione() { ...}	function nomefunzione() { ...}
formato ottale: "\377"	formato esadecimale: "\xff"

Tabella 12.2: Elenco di bashismi tipici

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.

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 12.3: Elenco di parametri di shell

Le nozioni base da ricordare riguardanti la **espansione dei parametri** sono le seguenti.

forma della espressione con parametri	valore se var è impostata	valore se var non è impostata
\${var:-stringa}	"\$var"	"stringa"
\${var:+stringa}	"stringa"	"null"
\${var:=stringa}	"\$var"	"stringa" (ed esegue "var=stringa")
\${var:?string}	"\$var"	invia con echo "stringa" allo stderr (ed esce con stato di errore)

Tabella 12.4: Elenco di espansioni di parametri di shell

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**

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 12.5: 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 [ <espressione_condizionale> ]; then
  <se_successo_esegue_questo_comando>
else
  <se_non_successo_esegue_questo_comando>
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>	<code><file></code> esiste
<code>-d <file></code>	<code><file></code> esiste ed è una directory
<code>-f <file></code>	<code><file></code> esiste ed è un file regolare
<code>-w <file></code>	<code><file></code> esiste ed è scrivibile
<code>-x <file></code>	<code><file></code> esiste ed è eseguibile
<code><file1> -nt <file2></code>	<code><file1></code> è più recente di <code><file2></code> (data di modifica)
<code><file1> -ot <file2></code>	<code><file1></code> è più vecchio di <code><file2></code> (data di modifica)
<code><file1> -ef <file2></code>	<code><file1></code> e <code><file2></code> sono sullo stesso device e stesso numero inode

Tabella 12.6: 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"`.

equazione	condizione perché venga restituito il valore logico "vero"
-z <str>	la lunghezza di <str> è zero
-n <str>	la lunghezza di <str> è diversa da zero
<str1> = <str2>	<str1> e <str2> sono uguali
<str1> != <str2>	<str1> e <str2> non sono uguali
<str1> < <str2>	se ordinate, <str1> viene prima di <str2> (dipendente dalla localizzazione)
<str1> > <str2>	se ordinate, <str1> viene dopo di <str2> (dipendente dalla localizzazione)

Tabella 12.7: Elenco di operatori per paragonare stringhe in espressioni condizionali

12.1.4 Cicli di shell

Ci sono diverse espressioni per cicli usabili nella shell POSIX.

- "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 [9.3.9](#).

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>
 - Metacaratteri: < > | ; & ()
 - 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
coreutils	V:886, I:999	15016	Utilità GNU di base
debianutils	V:938, I:999	213	utilità varie specifiche di Debian
bsdmainutils	V:864, I:998	586	raccolte di ulteriori utilità da FreeBSD
bsdutils	V:860, I:999	255	utilità di base per 4.4BSD-Lite
moreutils	V:4, I:22	201	utilità Unix aggiuntive

Tabella 12.8: Elenco di pacchetti contenenti piccoli programmi di utilità per script di shell

Suggerimento

Sebbene moreutils possa non esistere al di fuori di Debian, offre piccoli programmi interessanti. Quello più degno di nota è sponge(8) che piuttosto utile quando si desidera sovrascrivere il file originale.

pacchetto	popcon	dimensione	descrizione
x11-utils	V:247, I:635	637	xmessage(1): mostra un messaggio o richiesta in una finestra (X)
whiptail	V:425, I:996	69	mostra riquadri di dialogo amichevoli da script di shell (newt)
dialog	V:21, I:138	1119	mostra riquadri di dialogo amichevoli da script di shell (ncurses)
zenity	V:99, I:411	364	mostra riquadri di dialogo grafici da script di shell (gtk2.0)
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:66, I:361	6496	"/usr/bin/gettext.sh": traduce messaggi

Tabella 12.9: Elenco di programmi per interfaccia utente

12.1.7 Dialogo di script di shell

L'interfaccia utente di un semplice programma shell può essere migliorata dalla banale interazione con i comandi `echo` e `read` ad una più interattiva con l'uso dei cosiddetti programmi di dialogo ecc.

12.1.8 Esempio di script di shell con zenity

Questo è un semplice script che crea un'immagine ISO con dati RS02 forniti da `dvdisaster(1)`.

```
#!/bin/sh -e
# gmkrso2 : Copyright (C) 2007 Osamu Aoki <osamu@debian.org>, Public Domain
#set -x
error_exit()
{
    echo "$1" >&2
    exit 1
}
# Inizializza le variabili
DATA_ISO="$HOME/Desktop/iso-$$img"
LABEL=$(date +%Y%m%d-%H%M%S-%Z)
if [ $# != 0 ] && [ -d "$1" ]; then
    DATA_SRC="$1"
else
    # Seleziona la directory per creare l'immagine ISO dalla cartella sul desktop
    DATA_SRC=$(zenity --file-selection --directory \
        --title="Seleziona la radice dell'albero di directory per creare l'immagine ISO") \
        || error_exit "Uscita durante la selezione della directory"
fi
# Controlla la dimensione dell'archivio
xterm -T "Controllo dimensione $DATA_SRC" -e du -s $DATA_SRC/*
SIZE=$((du -s $DATA_SRC | awk '{print $1}'))/1024)
if [ $SIZE -le 520 ] ; then
    zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
        --text="La dimensione dei dati va bene per un CD di backup:\n $SIZE MB"
elif [ $SIZE -le 3500 ]; then
    zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
        --text="La dimensione dei dati va bene per un DVD di backup:\n $SIZE MB"
else
    zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
        --text="Dimensione dei dati troppo grande per farne il backup: $SIZE MB"
    error_exit "Dimensione dei dati troppo grande per farne il backup :\n $SIZE MB"
fi
# solo xterm sicuramente ha un'opzione -e funzionante
# Crea immagine raw ISO
rm -f "$DATA_ISO" || true
xterm -T "genisoimage $DATA_ISO" \
    -e genisoimage -r -J -V "$LABEL" -o "$DATA_ISO" "$DATA_SRC"
```

```
# Crea dati ridondanti RS02 supplementari
xterm -T "dvdisaster $DATA_ISO" -e dvdisaster -i "$DATA_ISO" -mRS02 -c
zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
  --text="dati ISO/RS02 ($SIZE MB) \n creati per: $DATA_ISO"
# EOF
```

Si potrebbe voler creare un lanciatore sul desktop con un comando definito in modo simile a `"/usr/local/bin/gmkrs02 %d"`.

12.2 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: [ prerequisiti ... ]
[ TAB ]   comando1
[ TAB ]   -comando2 # ignora errori
[ TAB ]   @comando3 # sopprime echo
```

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
<code>\$@</code>	target
<code>\$<</code>	primo prerequisito
<code>\$?</code>	tutti i prerequisiti più recenti
<code>^</code>	tutti i prerequisiti
<code>\$*</code>	<code>"%"</code> nome base con corrispondenza con il modello target

Tabella 12.10: 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 12.11: Elenco di espansioni delle variabili di `make`

Eseguire `"make -p -f/dev/null"` per vedere le regole interne automatiche.

12.3 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)

12.3.1 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 > esempio.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); /* previeni buffer overflow */
    y[10] = '\0'; /* riempi per assicurare che la stringa finisca con '\0' */
    printf("%5i, %5.3f, %10s, %10s\n", argc, x, y, argv[1]);
    return 0;
}
EOF
$ gcc -Wall -g -o eseg_esempio esempio.c -lm
$ ./eseg_esempio
    1, 2.915, ./run_exam,      (null)
$ ./eseg_esempio 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.

12.4 Debug

Il debug è un'importante fase del processo di programmazione. Sapere come fare il debug dei programmi rende buoni utenti Debian in grado di creare segnalazioni di bug significative.

12.4.1 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
```

Un buon tutorial su gdb viene fornito da "info gdb" o lo si può trovare [altrove in rete](#). Quello che segue è un piccolo esempio d'uso di gdb(1) su di un "programma" compilato con l'opzione "-g" per produrre informazioni di debug.

```
$ gdb program
(gdb) b 1                # imposta un punto di interruzione alla riga 1
(gdb) run argomenti      # esegue programma con argomenti
(gdb) next               # riga successiva
...
(gdb) step               # passo successivo
...
(gdb) p param            # stampa parametro
...
(gdb) p param=12         # imposta il valore a 12
...
(gdb) quit
```

Suggerimento

Molti comandi gdb(1) possono essere abbreviati. L'espansione del tasto di tabulazione funziona come nella shell.

12.4.2 Fare il debug di pacchetti Debian

Dato che tutti i binari installati in un sistema Debian dovrebbero essere, in modo predefinito, snelliti con strip, la maggior parte dei simboli di debug non è presente nei normali pacchetti. Per poter fare il debug di pacchetti Debian con gdb(1) devono essere installati i corrispondenti pacchetti *-dbg o *-dbgsym (ad esempio libc6-dbg per libc6, coreutils-dbg per coreutils).

I pacchetti nello stile vecchio forniscono il proprio pacchetto *-dbg corrispettivo. È messo direttamente nell'archivio principale Debian insieme al pacchetto originale stesso. I pacchetti più recenti possono generare automaticamente pacchetti *-dbgsym quando compilati e tali pacchetti di debug sono messi separatamente nell'archivio [debian-debug](#). Per maggiori informazioni fare riferimento agli [articoli sul Wiki Debian](#).

Se un pacchetto di cui si deve fare il debug non fornisce il proprio pacchetto *-dbg o *-dbgsym, è necessario installarlo dopo averlo ricompilato nel modo seguente.

```
$ mkdir /percorso/nuovo ; cd /percorso/nuovo
$ sudo apt-get update
$ sudo apt-get dist-upgrade
$ sudo apt-get install fakeroot devscripts build-essential
$ sudo apt-get build-dep nome_pacchetto_sorgente
$ apt-get source nome_pacchetto
$ cd nome_pacchetto*
```

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 nome_pacchetto*.changes
```

È necessario controllare gli script di compilazione del pacchetto ed assicurarsi di usare `"CFLAGS=-g -Wall"` per la compilazione di binari.

12.4.3 Ottenere un backtrace

Quando un programma va in crash, è una buona idea inviare un segnalazione di bug riportando le informazioni di backtrace.

Il backtrace può essere ottenuto eseguendo le azioni seguenti.

- Eseguire il programma in `gdb(1)`.
- Riprodurre il crash.
 - Ciò causa il ritorno al prompt di `gdb`.
- Digitare `"bt"` al prompt di `gdb`.

Nel caso in cui in programma si blocca, si può farlo andare in crash premendo `Ctrl-C` nel terminale in cui è in esecuzione `gdb`, ottenendo così il prompt di `gdb`.

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
```

12.4.4 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 12.12: Elenco di comandi gdb avanzati

12.4.5 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.

12.4.6 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 [9.3.6](#).

12.4.7 Strumenti per rilevazione di memory leak

In Debian sono disponibili svariati strumenti di rilevazione di memory leak.

pacchetto	popcon	dimensione	descrizione
libc6-dev	V:284, I:618	17379	<code>mtrace(1)</code> : funzionalità di debug malloc in glibc
valgrind	V:8, I:55	72070	strumento di debug e profilazione per la memoria
kmtrace	V:0, I:16	351	tracciatore per memory leak di KDE che usa <code>mtrace(1)</code> di glibc
alleyoop	V:0, I:1	655	frontend per Valgrind, programma di controllo della memoria
electric-fence	V:0, I:5	70	strumento di debug malloc(3)
leaktracer	V:0, I:2	56	tracciatore di memory leak per programmi C++
libdmalloc5	V:0, I:3	360	libreria per il debug dell'allocazione di memoria

Tabella 12.13: Elenco di strumenti per rilevazione di memory leak

12.4.8 Strumenti di analisi statica del codice

Esistono strumenti simili a [lint](#) per l'[analisi statica del codice](#).

12.4.9 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	descrizione
splint	V:0, I:4	2239	strumento per controllare staticamente la presenza di bug in programmi C
flawfinder	V:0, I:0	175	strumento per esaminare codice sorgente C/C++ e per cercare punti deboli per la sicurezza
perl	V:555, I:995	568	interprete con controllore interno statico del codice: B::Lint(3perl)
pylint	V:6, I:16	988	strumento di controllo statico del codice Python
weblint-perl	V:0, I:2	36	strumento di controllo della sintassi e dello stile di base per HTML
linklint	V:0, I:1	343	controllore veloce di collegamenti e strumento per manutenzione di siti web
libxml2-utils	V:22, I:311	167	utilità con xmllint(1) per convalidare file XML

Tabella 12.14: Elenco di strumenti per l'analisi statica del codice

12.5 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".

```
%o
Fabbrica componente Echo
```

In alternativa si può compilare con l'opzione per linker " -lflex" alla fine della propria riga di comando cc(1) (come "-ll" per AT&T-Lex). In questo caso non è necessario usare "%option".

12.6 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:10, I:110	2061	generatore GNU di parser LALR
byacc	V:0, I:7	160	generatore Berkeley di parser LALR
btyacc	V:0, I:0	207	generatore di parser backtracking basato su byacc

Tabella 12.15: 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.

```
%%
%%
```

12.7 Autoconf

[Autoconf](#) è uno strumento per produrre script shell che configurano automaticamente pacchetti software di codice sorgente, in modo da adattarsi a molti tipi di sistemi *nix usando l'intero sistema di compilazione GNU.

autoconf(1) produce lo script di configurazione "configure". "configure" crea automaticamente un "Makefile" personalizzato usando il modello "Makefile.in".

12.7.1 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
$ make install # questo mette i file nel sistema
```

12.7.2 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 "tutte-le-opzioni-che-erano-state-usate"
# 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
```

Se non si è sicuri di dove siano installati i file, si dovrebbe prendere in considerazione l'uso di checkinstall(8) dal pacchetto checkinstall, che fornisce un percorso di disinstallazione pulita. Ora permette di creare un pacchetto Debian con l'opzione `-D`.

12.8 Pazzie con corti script Perl

Benché qualsiasi script [AWK](#) possa essere riscritto automaticamente in [Perl](#) usando [a2p\(1\)](#), gli script AWK di una sola riga si convertono meglio manualmente in script Perl di una riga.

Si consideri il seguente pezzetto di script AWK.

```
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.

Per le opzioni per la riga di comando vedere `perlrun(1)`. Per altri script Perl pazzi può essere interessante guardare [Perl Golf](#).

12.9 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.

12.10 Traduzione di codice sorgente

Esistono programmi per convertire codice sorgente.

pacchetto	popcon	dimensione	parola chiave	descrizione
perl	V:555, I:995	568	AWK → PERL	converte codice sorgente da AWK a PERL:a2p(1)
f2c	V:0, I:8	433	FORTRAN → C	converte codice sorgente da FORTRAN 77 a C/C++: f2c(1)
intel2gas	V:0, I:0	174	intel → gas	convertitore da NASM (formato Intel) a GAS (GNU Assembler)

Tabella 12.16: Elenco di strumenti per la traduzione di codice sorgente

12.11 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 9.10.4 per capire le basi delle tecniche chroot
- `debuild(1)`, `pbuilder(1)` e `pdebuild(1)`
- Sezione 12.4.2 per la ricompilazione per il debug
- [Guida Debian per il nuovo manutentore](#) come tutorial (pacchetto `maint-guide`)
- [Debian Developer's Reference](#) (pacchetto `developers-reference`)
- [Manuale Debian Policy](#) (pacchetto `debian-policy`)
- [Guide for Debian Maintainers](#) (il pacchetto `debmake-doc`)

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)" 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.

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)"

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

Molte pagine di manuale e info del sistema Debian sono state usate come riferimento principale per la stesura di questo documento. Tanto che Osamu Aoki ha ritenuto nei limiti del [fair use](#) molte parti de esse, specialmente le definizioni dei comandi, che sono state riprese letteralmente dopo attenti sforzi editoriali per inserirle nello stile e nello scopo di questo documento.

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)" 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

I sorgenti del documento originale in inglese sono attualmente scritti usando file di testo [AsciiDoc](#); quest'ultimo è usato per motivi di comodità dato che richiede meno scrittura del puro XML e supporta le tabelle in un formato molto intuitivo. Si dovrebbero considerare i file XML e PO come i veri file sorgenti. Attraverso script di compilazione è convertito nel formato XML DocBook e i dati generati automaticamente sono inseriti a formare il sorgente XML Docbook finale. Questo sorgente finale può essere convertito in HTML, epub, testo semplice, PostScript e PDF. (Alcuni formati possono non essere distribuiti.)