Configurazione di base: rete, accounts, stampa ...

Capitolo 8

<<Lo scopo di una nuova installazione attraverso debian-installer è quello di rendere la macchina quanto più funzionante possibile, ma rimarranno comunque molti servizi da configurare. Inoltre, è sempre utile conoscere come modificare i diversi elementi di configurazione, definiti durante il processo d'installazione iniziale>>

Questo capitolo tratta tutto ciò che rientra in quello che potremo definire una "configurazione di base": networking, lingua ed i "locales" [il "locale", in informatica, è un insieme di parametri impostati dall'utente che riguardano la lingua, la regione ed eventuali preferenze speciali in merito all'interfaccia utente.], utenti e gruppi, stampa, mount points, ecc.

8.1. Come configurare il Sistema per un'altra lingua

Se il sistema è stato installato utilizzando la lingua francese, è probabile che sulla macchina sia già impostata come lingua predeflnita il francese. Ma è meglio conoscere come l'installer imposta la lingua in modo che possiate apportare modifiche in caso di necessità.

STRUMENTI TOOLS	
Il comando `locale` per visualizzare la configurazione corrente	Il comando locale elenca, attraverso un riepilogo della configurazione corrente, i diversi parametri “locales” (formato data, numbers format, ecc.), presentandoli sotto forma di un insieme di “variabili di ambiente standard” dedicate alla “modifica dinamica” di queste impostazioni. [Una variabile di ambiente è un valore con nome dinamico che può influire sul funzionamento dei processi in esecuzione su un computer. La “modifica dinamica”, possibile grazie a delle regole ed a dei criteri (campi chiave), è una modifica ex post ad una pregressa analisi e ad un campionamento dei dati, che ha per obiettivo un determinato “livello di qualità” e di conseguenza consente una reazione flessibile ai summenzionati dati raccolti.]

8.1.1. Come impostare la lingua predefinita

Un "locale" è un'insieme di impostazioni regionali [internazionali]. Tale insieme non include solo la lingua del testo, bensì anche il formato di visualizzazione dei numeri, delle date e degli orari e delle valute monetarie [per eventuali calcoli o conversioni], nonché la modalità di confronto alfabetica (per tenere conto propriamente dei caratteri accentati). Sebbene ciascuno di questi parametri possa essere specificato indipendentemente dagli altri, generalmente avrete necessità di utilizzare solo "un locale", in quanto di per sé un insieme compatto di valori per questo tipo di parametri che corrispondono a loro volta ad una "regione" in senso ampio. I "locales" sono generalmente indicati nel formato code-language_CODE-COUNTRY con talvolta un suffisso per specificarne il set di caratteri e la codifica che verrano impiegati se li selezionerete. Ciò consente di tenere conto delle differenze idiomatiche o tipografiche tra le diverse regioni con un linguaggio comune.

CULTURA	
Set di caratteri	Ogni locale è storicamente associato ad un “set di caratteri” (ovvero un insieme di caratteri noti) e ad una prescelta “codifica” (la codifica è il modo in cui sono rappresentati i caratteri nel computer). Le codifiche più popolari per le lingue latine erano limitate a 256 caratteri in quanto utilizzavano un byte per carattere. Dal momento che questa limitazione a 256 caratteri non consentiva di coprire sufficientemente tutte le lingue Europee, furono necessarie molteplici codifiche e così giunsero fra noi, in mezzo alle altre, dall’ISO-8859-1 (nota come “Latin 1”) sino all’ISO-8859-15 (“Latin 9”). Gli incarichi con lingue straniere, spesso implicavano regolari switches fra le codifiche ed i sets di caratteri. Inoltre, la redazione di documenti multilingue talvolta comportava problemi quasi insormontabili. Unicode (un super catalogo di quasi tutti i sistemi di scrittura linguistica nel mondo) è stato creato per aggirare questo problema. La sua speciale codifica UTF-8 conserva tutti i 128 simboli ASCII (con codifica a 7 bit), ma gestisce gli altri caratteri in modo differente. Questi sono preceduti da una specifica sequenza escape di pochi bits, che implicitamente definisce la lunghezza del carattere. [Una sequenza di escape è una combinazione di caratteri e ha un significato diverso dai caratteri letterali in essa contenuti; è contrassegnata da uno o più caratteri precedenti (e possibilmente terminanti)]. Ciò consente di codificare tutti i caratteri Unicode su uno o più bytes. L’uso di UTF-8 è diventato molto diffuso, per via del fatto che si tratta della codifica predefinita dei documenti XML. Questa codifica dovrebbe essere quella generalmente usata, difatti è di default sui sistemi Debian.

Il pacchetto `locales` include gli elementi necessari per il corretto funzionamento delle "localizzazioni" delle diverse applicazioni. Durante l'installazione, questo pacchetto pone alcune domande per selezionare il set delle lingue supportate. Questo set può essere cambiato in qualsiasi momento eseguendo da root `dpkg-reconfigure locales`.

La prima domanda vi invita a scegliere i "`locales`" da supportare. La selezione di tutti i locales inglesi (ovvero quelli che iniziano con "`en_`") è una scelta ragionevole. Non esitate a selezionare altri locales se la macchina ospita utenti stranieri. L'elenco di locales abilitati sul sistema è memorizzato nel file `/etc/locale.gen`. Potrete modificare manualmente questo file, ma dovrete ricordare di eseguire `locale-gen` dopo ogni modifica. `locale-gen` genererà i files necessari per il corretto funzionamento dei locales aggiunti, eliminando i files obsoleti.

La seconda domanda, intitolata "Default locale for the system environment", richiede di impostare un locale come default. La scelta raccomandata per gli utenti USA è "en_US.UTF-8". Gli utenti dell'Inghilterra potranno fare riferimento a "en_GB.UTF-8". Per la Francia è "fr_FR.UTF-8". I francofoni potranno scegliere: del Belgio "fr_BE.UTF-8"; del Lussemburgo "fr_LU.UTF-8", della Svizzera "fr_CH.UTF-8"; ed i canadesi "fr_CA.UTF-8" o "en_CA.UTF-8" (se anglofoni). Il file `/etc/default/locale` sarà quindi modificato per conservare la locale scelta. Dopodiché questo file sarà ripreso da tutte le sessioni-utente dal momento che il modulo PAM effettuerà un'injection del suo stesso contenuto nella variabile di ambiente LANG. [Un pluggable authentication module (PAM) è un meccanismo per integrare più schemi di autenticazione di basso livello in un'API (Application Programming Interface) di alto livello. Ciò consente ai programmi che si basano sull'autenticazione di essere scritti indipendentemente dallo schema di autenticazione sottostante].

DIETRO LE QUINTE

`/etc/environment` e `/etc/default/locale`

Il file `/etc/environment` supporta i programmi `login`, `gdm` o lo stesso `ssh` per la creazione delle corrette variabili d'ambiente. Queste applicazioni non creano le variabili direttamente, ma tramite un modulo PAM (pam_env.so). PAM (Pluggable Authentication Module) è una libreria modulare che centralizza i meccanismi di autenticazione, session initialization e password management. Andate a leggere il paragrafo 11.7.3.2 "Come configurare PAM" a pag. 315 per avere un esempio sulla configurazione di PAM.

Il file `/etc/defualt/locale` funziona allo stesso modo, ma contiene solo la variabile di ambiente LANG. Grazie a questa separazione alcuni utenti PAM possono ereditare un ambiente senza localizzazione. In realtà non è consigliabile eseguire i programmi server con la localizzazione abilitata; mentre si raccomandano la localizzazione e le impostazioni "regionali" [internazionali] per i programmi che aprono le sessioni utente.

8.1.2. Come configurare la tastiera

Sebbene il layout della tastiera sia gestito in modo diverso nella modalità grafica rispetto alla console, Debian fornisce un'unica interfaccia di configurazione che funziona per entrambe: questa interfaccia è basata su Debconf ed è implementata dal pacchetto `keyboard-configuration`. Pertanto il comando `dpkg-reconfigure keyboard-configuration` può essere utilizzato in qualsiasi momento per riconfigurare il layout della tastiera.

Le domande riguardano il layout della tastiera fisica (una tastiera per PC standard negli USA sarà una "Generic 104 Key" mentre in Francia sarà una "Generic 105 Key - Intl"), quindi in merito al layout da scegliere (generalmente US negli USA), e poi sulla posizione del tasto AltGr (right Alt). Infine viene posta una domanda sulla posizione da utilizzare per il "tasto Compose", che consente di inserire caratteri speciali combinando singoli caratteri. Ad esempio digitando consecutivamente il tasto "Compose" ed il tasto "e" produrrete un accento acuto ("é"). Tutte queste combinazioni sono

descritte nel file `/usr/share/X11/locale/en_US.UTF-8/Compose` (o in un altro file, determinato in base al "locale" corrente indicato da `/usr/share/X11/locale/compose.dir`).
Occorre precisare che la configurazione della tastiera per la modalità grafica qui descritta influisce solo sul layout predefinito; gli ambienti desktop GNOME e KDE Plasma, così come gli altri, forniscono un pannello di controllo della tastiera nelle loro preferenze, consentendo a ciascun utente di avere il proprio layout. Alcune opzioni aggiuntive riguardanti il comportamento di alcuni tasti particolari sono disponibili anche nei suddetti pannelli di controllo.

8.1.3. Migrazione a UTF-8

La standardizzazione della codifica UTF-8 è stata una soluzione molto attesa per risolvere le molteplici problematiche di interoperabilità, poiché incentiva lo scambio [di dati] internazionale e rimuove i limiti decisionali in merito ai caratteri che possono essere utilizzati in un documento.
L'aspetto negativo è che abbiamo dovuto attraversare una fase di conversione piuttosto difficoltosa. Dal momento che rischiava di non essere completamente cristallina (o perlomeno non sincronizzata in tutto il mondo), furono necessarie due operazioni di conversione: una sui contenuti dei files, una sui filenames. Fortunatamente, la maggior parte della suddetta migrazione è stata completata e viene argomentata meramente come citazione.

CULTURA
Mojibake e gli errori di interpretazione

Se un testo viene trasmesso (o memorizzato) senza “encoding information”, non è detto che il destinatario sia in grado di individuare con certezza quale convenzione utilizzare per interpretare il significato di ciò che in fin dei conti è un insieme di bytes. [Genericamente “l’econding information” è un’informazione che individua univocamente la codifica applicata]. Di solito è possibile farsi un’idea attraverso un’analisi statistica sulla distribuzione dei valori presenti nel testo, ma questo non vi darà una risposta certa. Quando lo enconding system scelto per la lettura differisce da quello utilizzato per la scrittura, i bytes vengono interpretati in modo errato e nella migliore delle ipotesi si ottengono degli errori in alcuni caratteri, mentre nel peggiore dei casi qualcosa di completamente illeggibile.
Pertanto, se un testo in francese è apparentemente normale ad eccezione delle lettere accentate e di alcuni simboli, che sono stati sostituiti da sequenze di caratteri del tipo “Ã©”,“Ã¨” o “Ã§”, è probabilmente un file codificato in UTF-8 ma interpretato come ISO-8859-1 o ISO-8859-15. Ciò significa che l’installazione locale non è ancora stata migrata a UTF-8. Se, d’altra parte, vengono visualizzati dei punti interrogativi al posto delle lettere accentate - oppure se i suddetti punti interrogativi hanno apparentemente sostituito un carattere che avrebbe dovuto seguire la summenzionata lettera accentata, è probabile che l’installazione sia già stata configurata in UTF-8 e che vi è stato inviato un documento codificato in ISO-8859-*.
Quanto espresso riguarda i casi “semplici”. Questi casi si verificano solo per le culture occidentali, perché Unicode (e UTF-8) è stato progettato per massimizzare i punti comuni delle codifiche storiche per le lingue occidentali basate sull’alfabeto latino, consentendo il riconoscimento delle parti del testo in cui i caratteri sono mancanti.
Diversamente nelle configurazioni più complesse, dove ad esempio due ambienti sono associati a due lingue diverse che non usano lo stesso alfabeto, otterrete spesso dei risultati illeggibili - una successione di simboli astratti che non hanno nulla a che fare con le due suddette lingue. Questa situazione è particolarmente frequente in Asia, a causa delle molteplicità delle lingue e dei sistemi di scrittura. La parola giapponese mojibake è stata adottata per designare questo fenomeno.
Quando si verifica, la diagnostica è più complessa e la soluzione più semplice è spesso migrare a UTF-8 in entrambi gli ambienti.

Riguardo ai filenames, la migrazione può essere relativamente semplice. Il tool `convmv` (contenuto nel pacchetto omonimo) è stato appositamente scritto per questo scopo; consente di rinominare i

files da una codifica all'altra. L'uso di questo strumento è relativamente semplice, ma vi consigliamo di farlo in due fasi per evitare sorprese. L'esempio seguente mostra un ambiente UTF-8 che contiene i nomi delle directories codificati in ISO-8859-15 e come utilizzare `convmv` per la loro ridenominazione.

```
$ ls travail/
Ic?nes ?1?ments graphiques Textes
$ convmv -r -f iso-8859-15 -t utf-8 travail/
Starting a dry run without changes...
mv "travail/lments graphiques" "travail/Éléments graphiques"
mv "travail/Icnes" "travail/Icônes"
No changes to your files done. Use  --notest to finally rename the files.
$ convmv -r --notest -f iso-8859-15 -t utf-8 travail/
mv "travail/lments graphiques" "travail/Éléments graphiques"
mv "travail/Icnes" "travail/Icônes"
Ready!
$ ls travail/
Éléments graphiques Icónes Textes
```

Per il contenuto dei files, la procedura di conversione sarà più complessa, a causa della molteplicità dei formati di files esistenti. Alcuni formati dei files contengono l'encoding information, agevolando le attività del software che li elabora; sarà sufficiente poi aprire quei files e salvarli di nuovo, specificando la codifica UTF-8. In altri casi, sarà necessario specificarne la codifica originale (ISO-8859-1 o "Western", oppure ISO-8859-15 o "Western (Euro)" a seconda delle formulazioni) all'apertura del file.

Per i semplici files di testo, potrete utilizzare `recode` (contenuto nell'omonimo pacchetto), che consente la ricodifica automatica. Questo strumento ha molteplici opzioni che consentono di giocare con il suo funzionamento. Vi invitiamo a consultare la sua documentazione, la `recode (1)` man page o la `recode` info page (più completa).

8.2. Come configurare la rete

BASILARE
Rete: concetti essenziali (Ethernet, indirizzo IP, subnet - sottorete, broadcast ...)

La maggior parte delle moderne reti locali utilizzano il protocollo Ethernet, in cui i dati sono suddivisi in blocchi chiamati frames e sono trasmessi sul cavo un frame alla volta. La velocità della trasmissione dei dati varia da 10 Mbit/s per le schede Ethernet più vecchie, a 10 Gbit/s per le schede di generazione più recente (al momento della stesura di questo testo il valore più frequente si aggira dai 100 Mbits a 10 Gb/s). I cavi più comuni sono chiamati in base alla capacità che possano supportare in modo affidabile (la T sta per “twisted pair” o doppino): 10BASE-T, 100BASE-T, 1000BASE-T o 10GBASE-T e 40GBASE-T; ciascuno di questi cavi è dotato di un connettore RJ45. Esistono altri tipi di cavi, che vengono utilizzati principalmente per la velocità di trasmissione dati da 10 Gb/s in su.
Un indirizzo IP è un numero utilizzato per identificare un’interfaccia di rete su un computer di una rete locale o su Internet. Nella versione attualmente più recente di IP (IPv4), questo numero è codificato a 32 bits ed è generalmente rappresentato con 4 numeri separati da punti (es: 192.168.0.1), ogni numero può variare da 0 a 255 (255 incluso, che corrisponde a 8 bit di dati). La versione susseguente del suddetto protocollo, il protocollo IPv6, estende “l’address space” [“l’address space” è un intervallo di indirizzi discreti, ciascuno dei quali può corrispondere ad un host di rete, un dispositivo periferico, un settore del disco, una cella di memoria o un’altra entità logica o fisica] a 128 bits, e gli indirizzi sono rappresentati come numeri esadecimali separati da due punti (es: `2001:0db8:13bb:0002:0000:0000:0020`, che può essere abbreviato in `2001:db8:13bb:2::20`).

Una maschera di sottorete (subnet mask - netmask) definisce attraverso la sua stessa codifica binaria quale parte di un indirizzo IP corrisponde alla rete, mentre la parte rimanente specifica l'identificatore della macchina.
Ad esempio per la configurazione di una data IPv4 statica, la maschera di sottorete 255.255.255.0 (ventiquattro "1" seguiti da otto "0" in rappresentazione binaria) indica che i primi 24 bit dell'indirizzo IP corrispondono all'indirizzo di rete, mentre gli ultimi 8 appartengono al numero della macchina. In IPv6, per motivi di leggibilità, è espresso solo con un numero composto da "1". Di conseguenza una maschera di rete IPv6 è di 64 bit.
L'indirizzo di rete è un indirizzo IP la cui parte che descrive il numero della macchina è zero. L'intervallo di indirizzi IPV4 di un'intera rete è spesso descritto dalla sintassi `a.b.c.d/e` dove a.b.c.d è l'indirizzo di rete ed e il numero di bit assegnato al ruolo (nella rete) in un indirizzo IP. Un indirizzo di rete ad esempio potrebbe quindi essere: 192.168.0.0/24. La sintassi è simile con il protocollo IPv6: 2001:db8:13bb:2::/64.
Un router è una macchina che collega diverse reti insieme. Tutto il traffico che arriva attraverso il router viene reindirizzato alla rete corretta. Per fare ciò, il router analizza i pacchetti in arrivo e li reindirizza in base all'indirizzo IP della loro destinazione. Il router viene spesso definito gateway e funziona come una macchina che consente di uscire da una rete locale (verso un extended network come lo stesso Internet).
Lo special broadcast address [La IETF e la IANA hanno limitato dall'uso generale vari indirizzi IP riservati a scopi speciali tra cui il broadcast address] consente di raggiungere tutte le stazioni di rete. Quasi mai "routed", tale indirizzo funziona solo sulla rete in questione. Concretamente, ciò significa che un pacchetto dati indirizzato ad un indirizzo broadcast non oltrepasserà mai un router.
Si precisa che in questo capitolo vengono trattati gli indirizzi IPv4, più comunemente utilizzati al momento della stesura di questo testo. Il protocollo IPv6 sarà discusso dettagliatamente nel paragrafo 10.6, "IPv6" a pagina 257, ma i concetti essenziali rimangono gli stessi.

La rete viene configurata automaticamente durante l'installazione iniziale. Se installerete anche Network Manager (use-case tipico delle installazioni full per desktop) non dovrete effettuare alcuna configurazione (ad esempio non dovrete definire alcuna specifica qualora utilizziate un DCHP server ed una connessione cablata). In pratica se è necessaria una connessione (ad esempio tramite un'interfaccia WiFi) Network Manager creerà il file pertinente in `/etc/NetworkManager/system-connections/` .

NOTA
NetworkManager

Network Manager è particolarmente indicato per le configurazioni roaming (andate a leggere il paragrafo 8.2.5, "Configurazione della rete automatica per i Roaming Users" a pagina 169), ma può essere utilizzato anche come strumento predefinito per la gestione della rete. Potrete realizzare delle "System connections", che vengono attivate all'avvio del computer, creando manualmente un file `.ini-like` in `/etc/NetworkManager/system-connections/`, o tramite uno strumento grafico (`nm-connection-editor`). Dovrete solo ricordare di disabilitare tutte le voci in `/etc/network/interfaces` affinché Network Manager si occupi delle suddette interfacce.

- https://wiki.gnome.org/Projects/NetworkManager/SystemSettings
- https://developer.gnome.org/NetworkManager/1.14/ref-settings.html

Diversamente se non è installato Network Manager l'installer configura `ifupdown` attraverso la creazione del file `/etc/network/interfaces`. Una riga che inizia con `auto` fornisce l'elenco delle

interfacce che saranno configurate automaticamente all'avvio dai servizi di rete [`ifupdown` e dal suo script init `/etc/init.d/networking`.] Se disponete di diverse interfacce di rete dovreste mantenere la loro configurazione in files distinti nella directory `/etc/network/interfaces.d/` . `ifupdown` è lo strumento per configurare la rete solitamente utilizzato nei servers. Per tale ragione verrà trattato nei capitoli a seguire

8.2.1 Interfaccia Ethernet

Se il computer ha una scheda di rete Ethernet, è necessario configurare la rete IP ad essa associata attraverso uno dei seguenti (due) metodi. Il più semplice dei due metodi è la configurazione dinamica tramite DHCP, che richiede la presenza di un server DHCP sulla rete locale.
Potrete indicare un nome host a piacere, che corrisponde al parametro opzionale hostname nell'esempio sottostante. Il server DHCP quindi vi restituirà i parametri di configurazione di rete appropriati.

Esempio 8.1 Configurazione DHCP

```
auto enp0s31f6
iface enp0s31f6 inet dhcp
  hostname arrakis
```

IN PRATICA

La denominazione delle interfacce di rete

Per impostazione predefinita, il kernel attribuisce dei nomi generici alle interfacce di rete (ad esempio `eth0` per l'interfaccia Ethernet e `wlan0` per l'interfaccia Wi-Fi). Il carattere numerico è un contatore incrementale e viene assegnato all'interfaccia di rete in base al suo rilevamento (ed alla sua posizione nell'ordine di rilevamento). Pertanto il nome predefinito dell'interfaccia di rete non è affidabile nelle macchine moderne dato che cambia ad ogni riavvio in base all'ordine di rilevamento. Fortunatamente `systemd` ed `udev` assegnano i nomi all'interfacce automaticamente al loro rilevamento. La policy sui nomi predefiniti (`NamePolicy`) viene definita da `/lib/systemd/network/99-default.link` (per maggiori informazioni sull'entry `NamePolicy` fate riferimento alla man page `systemd.link(5)`). In pratica i nomi sono assegnati in base alla posizione fisica dei devices (ovvero in base a dove vengono connessi) ed al tipo di interfaccia di rete, `en` se ethernet e `wl` se Wi-Fi. Nell'esempio soprastante il nome `enp0s31f6` è stato assegnato all'interfaccia di rete in base alla seguente logica: `en` (Ethernet), `p` (PCI), `0` (bus number), `s31` (slot number), `f6` (function number).
Ovviamente potete `override` (trad. lett. ignora o ignora e sostituisci con) la suddetta policy oppure integrarla personalizzando i nomi di alcune interfacce specifiche. Inoltre potrete rintracciare, sotto forma di output, i nomi delle interfacce di rete attraverso `ip addr` (oppure come filenames inclusi in `/sys/class/net`).

Quanto appena espresso potrebbe servirvi qualora insorgesse un `corner case` [o `pathological case` - in ingegneria un contesto che si verifica al di là dei normali parametri operativi] per cui dovrete disabilitare i nomi dei dispositivi di rete assegnati dalla policy. Per raggiungere tale obiettivo potrete inoltre modificare la regola predefinita `udev` per avviare il sistema utilizzando i parametri kernel `net.ifnames=0` e `biosdevname=0`.

Per una configurazione "statica" dovrete indicare i parametri di rete in modo statico. Per fare ciò dovrete includere almeno l'indirizzo IP e la maschera di sottorete; a volte nell'elenco della configurazione statica sono inclusi anche il network address ed il broadcast address. Un router che consente delle connessioni al di fuori della rete verrà menzionato come `gateway`.

Esempio 8.2 Configurazione Statica

```
auto enp0s31f6
iface enp0s31f6 inet static
  address 192.168.0.3/24
  broadcast 192.168.0.255
  network 192.168.0.0
  gateway 192.168.0.1
```

NOTA
Indirizzi multipli

È possibile non solo associare più interfacce ad una singola scheda di rete fisica, ma anche diversi indirizzi IP ad una singola interfaccia. Inoltre si precisa che un indirizzo IP può corrispondere tramite DNS a diversi nomi e che un nome può a sua volta corrispondere a diversi indirizzi IP numerici.
Comprenderete che le configurazioni possono essere molto complesse, ma le summenzionate opzioni vengono utilizzate solo in casi molto specifici. Gli esempi citati qui riguardano solo le configurazioni ordinarie.

8.2.2 Interfaccia Wireless

L'installazione delle interfacce di rete wireless potrebbe rivelarsi impegnativa. Difatti la scheda di rete potrebbe richiedere l'installazione di un firmware proprietario non installato automaticamente da Debian. Inoltre le schede di rete wireless si basano su un sistema di crittografia per limitare l'accesso di rete solo agli utenti autorizzati e pertanto dovrete salvare nella configurazione di rete anche una chiava segreta. Tali argomenti verranno affrontati a seguire passo dopo passo.

8.2.2.1 Installazione dei firmwares

Innanzitutto dovrete abilitare il repository APT non-free nel file `sources.list`: per maggiori informazioni al riguardo andate a leggere il paragrafo 6.1 Come compilare il file `sources.list` a pag. 108. Diversi firmware si trovano nel repository non-free in quanto rilasciati con licenza proprietaria. Potete tentare di saltare questa fase, ma se il firmware risultasse "mancante" dovrete comunque tornare sui vostri passi ed abilitare il repository APT non-free.

Se non siete in grado di risalire al pacchetto di cui necessitate potete provare ad installare `isenkram` ed eseguire successivamente il comando `isenkram-autoinstall-firmware`. La denominazione dei pacchetti deriva spesso dal nome del produttore o dal nome del modulo kernel corrispondente, ad esempio: `firmware-iwlwifi` (Intel Wireless Cards), `firmware-atheros` (Qualcomm Atheros), `firmware-ralink` (Ralink), ecc.. Dovrete poi effettuare il riavvio del sistema in quanto il kernel driver rileva i files del firmware soltanto durante il suo caricamento iniziale e non successivamente.

8.2.2.2 Alcune voci specifiche per le reti wireless in /etc/network/interfaces

`ifupdown` è in grado di gestire le interfacce wireless attraverso l'ausilio del pacchetto `wpasupplicant` che supporta l'integrazione necessaria del comando `ifupdown` e del comando `wpa_supplicant` allo scopo di configurare le interfacce wireless (se utilizzano una tecnologia di criptazione `WPA/WPA2`). Inoltre dovrete aggiungere alla classica entry in `/etc/network/interfaces` due parametri supplementari per definire il nome della rete wireless (`SSID`) e la `Pre-Shared Key` (PSK).

Esempio 8.3 Configurazione DHCP per un'interfaccia wireless

```
auto wlp4s0
iface wlp4s0 inet dhcp
 wpa-ssid Falcot
 wpa-psk ccb290fd4fe6b22935cbae31449e050edd02ad44627b16ce0151668f5f53c01b
```

Il parametro `wpa-psk` può includere sia la passphrase in `plain text` o la sua versione in hash generata con `wpa_passphrase SSID passphrase`. Se utilizzate connessioni wireless non criptate potrete immettere `wpa-key-mgmt NONE` e non includere alcuna voce `wpa-entry`. Per maggiori informazioni sulle opzioni di configurazioni disponibili consultate `/usr/share/doc/wpasupplicant/README.Debian.gz`.

Infine dovreste considerare di limitare i permessi di lettura di `/etc/network/interfaces/` soltanto all'utente di root in modo da impedire agli utenti generici l'accesso alla chiave privata.

STORIA Criptazione WEP	Potrete usufruire della tecnologia deprecata WEP attraverso il pacchetto `wireless-tools`. Se cercate istruzioni su come procedere consultate `/usr/share/doc/wireless-tools/README.Debian`.

8.2.3 La connessione PPP tramite modem PSTN

Una connessione point to point (PPP) stabilisce una intermittent connection; questa soluzione è la più diffusa per le connessioni con modem-telefono (tale tipo di modem viene definito "modem PSTN", dal momento che la connessione avviene tramite la Public Switched Telephone Network o Rete Telefonica Generale - RTG o Rete Telefonica Pubblica Commutata - PSTN).

Una connessione tramite modem-telefono richiede un account con un access provider ed inoltre un numero di telefono, un nome utente, una password e, talvolta, un authentication protocol. Essendo di fatto una connessione viene configurata attraverso un tool, il `pppconfig` , che è incluso nell'omonimo pacchetto Debian.

Per impostazione predefinita, viene configurato un "connection name" (che corrisponde a quello dell'Internet service provider). In caso di dubbi sull'authentication protocol, selezionare `PAP` [Password Authentication Protocol]: è supportato dalla maggior parte degli Internet service providers.

Dopo la configurazione, è possibile connettersi utilizzando il comando `pon` (seguito, sotto forma di parametro, dal nome della connessione qualora il valore predefinito `provider` non sia idoneo). Per chiudere la connessione utilizzate il comando `poff`. Questi due comandi possono essere eseguiti dall'utente root o da qualsiasi altro utente, che faccia parte del gruppo `dip`.

8.2.4 La connessione tramite modem ADSL

Il termine generico "modem ADSL" indica svariati devices con funzioni diverse fra loro. I modems più semplici da usare con Linux sono quelli che hanno un'interfaccia Ethernet (e non solo un'interfaccia USB). Quest'ultimi si stanno diffondendo considerevolmente; la maggior parte degli ADSL Internet service providers prestano (o noleggiano) un "box" con interfacce Ethernet. A seconda del tipo di modem, la configurazione richiesta può variare notevolmente.

8.2.4.1 I Modems che supportano PPPOE

Alcuni modem Ethernet funzionano con il protocollo PPPOE (Point to Point Protocol over Ethernet). Il tool `pppoeconf` (che deriva dal pacchetto omonimo) sarà in grado di configurare la connessione. Per fare ciò, modificherà il file `/etc/ppp/peers/dsl-provider` con i parametri supportati e salverà i dati per il login nei files `/etc/ppp/pap-secrets` e `/etc/ppp/chap-secrets`. Vi raccomandiamo di accettare tutte le modifiche che vi propone.

Quando avrete ultimato questa configurazione, potrete avviare la connessione ADSL con il comando `pon dsl-provider` ed interromperla con il comando `poff dsl-provider`.

SUGGERIMENTO
Come eseguire ppp all'avvio del computer

Le connessioni PPP tramite ADSL sono, per definizione, intermittenti. Dato che tali connessioni non sono fatturate a consumo e che ci sono pochi aspetti negativi nell'approfittarne, sarete tentati a mantenere sempre aperta la connessione; un metodo semplice per fare quanto sopra espresso è utilizzare il processo init.
Con `systemd` per aggiungere una task che riavvi automaticamente la connessione ADSL, è sufficiente creare, un "file unit", [systemd salva le istruzioni di inizializzazione di ciascun demone in un correlato file di configurazione, denominato "file unit"], come ad esempio `/etc/systemd/system/adsl-connection.service` con un contenuto simile al seguente:

```
[Unit]
Description=ADSL connection

[Service]
Type=forking
ExecStart=/usr/sbin/pppd call dsl-provider
Restart=always

[Install]
WantedBy=multi-user.target
```

Una volta che avrete configurato il file unit, dovrete abilitarlo con il comando `systemctl enable adsl-connection`. Dopodiché potrete avviare il loop ["loop" in inglese significa letteralmente "ciclo continuo"] con il comando `systemctl start adsl-connection`; così facendo tale processo si avvierà automaticamente durante il boot.
Sui sistemi che non usano systemd (inclusi quelli che eseguono Wheezy o altre versioni antecedenti di Debian), il processo predefinito SystemV funziona in modo diverso. Su tali sistemi, dovrete semplicemente aggiungere la seguente riga alla fine del contenuto del file `/etc/inittab`; in questo modo tutte le volte che la connessione dovesse interrompersi il processo init la riavvierà.

```
adsl:2345:respawn:/usr/sbin/pppd call dsl-provider
```

Per le connessioni ADSL che subiscono una disconnessione automatica al giorno, questo metodo consente di ridurre la durata dell'interruzione.

8.2.4.2 I Modems che supportano PPTP

Il protocollo Point-to-Point Tunneling Protocol (PPTP) fu creato dalla Microsoft. Fu distribuito agli esordi dell'ADSL ed è stato poi rapidamente sostituito dal PPPOE. Qualora foste costretti ad utilizzare questo protocollo, andate a leggere il paragrafo 10.3.4, "PPTP" a pagina 250.

8.2.4.3 I Modems che supportano DHCP

Quando un modem è collegato al computer con un cavo Ethernet (un cavo crossover), potrete configurare sul suddetto computer la classica connessione di rete via DHCP; il modem funzionerà automaticamente come un gateway per impostazione predefinita e si occuperà del routing (ovvero della gestione del traffico di rete fra il computer e Internet). [Un cavo crossover è un tipo cavo che consente a due computer o due device di pari livello di essere collegati e di potersi scambiare dati reciprocamente].

BASILARE
Cavi crossover per una connessione diretta via Ethernet

Le schede di rete dei computers supportano la ricezione dei dati su degli specifici fili del cavo e trasmettono i loro dati attraverso gli altri.
Quando connettete un computer ad una rete locale, solitamente collegate un cavo (straight o crossover) tra la scheda di rete ed un ripetitore o uno switch. Tuttavia, se desiderate collegare direttamente due computer (vale a dire senza uno switch o un ripetitore intermedio), dovrete instradare il segnale dalla sezione trasmittente di una scheda alla sezione ricevente dell'altra e viceversa. Al summenzionato servizio ci pensa il cavo crossover, che è per l'appunto impiegato per tale ragione.
Si precisa che la suddetta distinzione è irrilevante, dato che le moderne schede di rete sono in grado di rilevare automaticamente il tipo di cavo utilizzato e di adattarsi di conseguenza; pertanto non è raro che entrambi i tipi di cavo funzionano in modo identico nella medesima ubicazione.

La maggior parte dei "routers ADSL" sul mercato funziona in questo modo, così come la maggior parte dei modems ADSL concessi dai fornitori di servizi Internet.

8.2.5. Configurazione della rete automatica per i Roaming Users

Molti ingegneri della Falcot hanno un computer laptop che usano sia per scopi professionali, sia a casa. La configurazione di rete da utilizzare differisce in base all'ubicazione. A casa, potrebbe essere una rete Wi-Fi (protetta da una chiave WPA), mentre al lavoro potrebbe essere una rete cablata che offre maggiore sicurezza e maggiore velocità.

Per evitare di dover attivare o disattivare manualmente le interfacce di rete correlate, gli amministratori hanno installato il pacchetto `network-manager` sulle suddette macchine roaming. Questo software consente all'utente di effettuare facilmente lo switch da una rete all'altra utilizzando una piccola icona dell'area di notifica del loro ambiente desktop. Un clic sull'icona attiva la

visualizzazione di un elenco di reti disponibili (cablate e Wi-Fi), di conseguenza gli amministratori dovranno soltanto scegliere la rete da utilizzare. Il programma salva le reti in cui l'utente ha già effettuato l'accesso e passa automaticamente alla migliore rete disponibile quando perde la connessione in corso.

A tale scopo, il programma è strutturato in due parti: un demone in esecuzione con permessi di root esegue le operazioni di attivazione e configurazione delle interfacce di rete mentre un'interfaccia utente controlla a sua volta questo demone. PolicyKit gestisce le autorizzazioni necessarie per controllare questo programma; Debian ha configurato PolicyKit in modo che solo i membri del gruppo `netdev` abbiano i permessi per creare o modificare le connessioni di Network Manager.

Network Manager in questo modo potrà gestire diversi tipi di connessione (DHCP, configurazione manuale, rete locale), ma a patto che la configurazione sia stata eseguita suo tramite. Questo è il motivo per cui ignorerà sistematicamente tutte le interfacce di rete in `/etc/network/interfaces` ed `/etc/network/interfaces.d` in non soddisfacenti. Dal momento che Network Manager non fornisce dettagli riguardo ad un'eventuale assenza di connessioni d rete disponibili, il modo più semplice per risolvere qualsiasi avversità è rimuovere da `/etc/network/interfaces` le eventuali configurazioni di tutte le interfacce di rete che devono essere gestite dallo stesso Network Manager.

Si precisa che questo programma viene installato per impostazione predefinita durante l'installazione iniziale come effetto della selezione della task denominata "Desktop Environment".

8.3 Assegnazione dell'Hostname e Configurazione del Name Service

Lo scopo dell'assegnazione dei nomi ai numeri IP è di renderli facili da memorizzare per gli esseri umani. In realtà, un indirizzo IP identifica un'interfaccia di rete correlata ad un dispositivo come ad esempio una scheda di rete. Dato che ciascuna macchina può avere diverse schede di rete, e diverse interfacce su ciascuna scheda, ogni singolo computer può avere a sua volta diversi nomi nel domain name system - DNS [Il Domain Name System - DNS (o sistema dei nomi di dominio in italiano) è un sistema utilizzato per assegnare i nomi agli hosts (o nodi della rete in italiano)].

Ogni macchina è comunque identificata da un prenome (o "main name" o "canonical"), conservato nel file `/etc/hostname` e comunicato al kernel Linux dagli scripts di inizializzazione attraverso il comando `hostname`. Il "valore corrente" corrispondente all'hostname è disponibile in un virtual filesystem e potrete reperirlo attraverso il comando `cat /proc/sys/kernel/hostname`.

BASILARE /proc/ e /sys/, filesystem virtuali	Gli alberi (o le strutture ad albero) dei files /proc/ e /sys/ sono generati dai file systems "virtuali". Questo metodo è pratico per recuperare informazioni del kernel (attraverso l'elenco dei files virtuali) e comunicarle allo stesso kernel (attraverso la redazione di files virtuali). `/sys/`in particolare consente l'accesso agli internal kernel objects, specialmente a quelli che rappresentano le varie periferiche del sistema. Il kernel in questo modo può condividere diversi tipi di informazione: lo stato di ciascun dispositivo (ad esempio, se è in modalità di risparmio energetico), se è rimovibile, ecc. Si precisa che `/sys/` è nato ed è stato reso disponibile a partire dalla versione 2.6. del kernel. `/proc/` descrive lo stato corrente del kernel: i files di questa directory contengono informazioni sui processi in esecuzione nel sistema e sull'hardware.

Sorprendentemente, il nome di dominio non è gestito allo stesso modo, ma deriva dal nome completo della macchina, ottenuto attraverso la name resolution. Potrete modificarlo nel file /etc/hosts; vi

basterà inserire un nome completo per la macchina all'inizio dell'elenco dei nomi associati agli indirizzi della macchina come da esempio seguente:

```
127.0.0.1 localhost
192.168.0.1 arrakis.falcot.com arrakis
```

8.3.1. Risoluzione dei nomi

Il sistema di risoluzione dei nomi di Linux è modulare e può fare affidamento su diverse sorgenti di informazione dichiarate nel file `/etc/nsswitch.conf`. La voce inerente alla risoluzione del nome dell'host è `hosts`. Per impostazione predefinita, quest'ultima voce elenca come servizi `files` e `dns`, il che significa che il sistema [seguendo l'ordine con cui sono stati elencati i servizi] consulta prima il file `/etc/hosts` e poi interroga i server DNS. I servers NIS/NIS+ o LDAP sono altre possibili sorgenti.

NOTA	
NSS e DNS	Fate attenzione, i comandi destinati specificatamente per il query DNS (specialmente `host`) non usano lo standard name resolution mechanism (NSS). Pertanto non tengono conto del file `/etc/nsswitch.conf`, per non parlare di `/etc/hosts`.

8.3.1.1 Configurazione del server DNS

[In generale il `mapping` è una tecnica di gestione dei dati (per poter eseguire una migrazione, integrazione, ecc), che consente l'estrazione dei dati dei `campi` (in inglese `fields`) di un database ed il loro indirizzamento ai campi corrispondenti di altri database targets. Pertanto, genericamente, i databases i cui dati vengono fra loro indirizzati attraverso questa tecnica possono essere definiti `mapping databases`]

Il DNS (Domain Name Service) è un servizio (distribuito e gerarchico) che si occupa del mapping dei nomi di dominio associati agli indirizzi IP e viceversa. In pratica è in grado di associare un nome facile da ricordare per gli esseri umani come `www.eyrolles.com` ad un concreto indirizzo IP come ad esempio 213.244.11.247.

Per accedere alle informazioni DNS, è necessario disporre di un server DNS che trasmetta le richieste. Anche se la Falcot Corp possiede un server DNS, generalmente un utente-tipo è più propenso ad usufruire dei servers DNS offerti dal suo ISP.

I servers DNS che verranno utilizzati sono specificati nel file `/etc/resolv.conf` ed elencati uno per riga, attraverso l'impiego della parola chiave `nameserver` che precede l'indirizzo IP, come mostrato nell'esempio seguente:

```
nameserver 212.27.32.176
nameserver 212.27.32.177
nameserver 8.8.8.8
```

Si precisa che il file /etc/resolv.conf potrebbe essere stato configurato automaticamente (e sovrascritto) se la rete è gestita da NetworkManager o è stata configurata attraverso DHCP.

8.3.1.2 I1 file /etc/hosts

Anche in assenza di un name server nella rete locale, potrete comunque definire una breve table che si occupa del mapping tra gli indirizzi IP e gli hostnames nel file `/etc/hosts`, di solito quest'ultimo riservato alle local network stations. La sintassi di questo file, descritta anche in `hosts(5)`, è molto semplice: ogni riga indica uno specifico indirizzo IP seguito dall'elenco di tutti i nomi ad esso associati (il primo nome host è "completely qualified", ovvero include il nome di dominio).

Questo file è disponibile sia quando la rete è fuori servizio, sia quando i server DNS sono irraggiungibili, ma potrà esservi davvero utile solo se lo duplicherete su tutti i computers della rete. Qualsiasi cambiamento minimo di questo file ne comporterà il suo aggiornamento obbligatorio su tutte le macchine della rete. Per questo motivo il file /etc/hosts di solito contiene solo le voci più importanti.
Per una piccola rete non connessa ad Internet questo file sarà sufficiente, ma da cinque macchine in sù si consiglia di installare un server DNS adeguato.

SUGGERIMENTO Come scavalcare il DNS	Poiché le applicazioni consultano il file /etc/hosts prima di interrogare ("querying") il DNS, è possibile inserire informazioni diverse rispetto a quelle che normalmente il DNS restituirebbe, scavalcando di fatto il normale sistema "DNS-based name resolution". Quanto sopra espresso vi consente, qualora le modifiche del DNS non siano state ancora "propagate", di testare l'accesso ad un sito Web con il nome previsto anche se non è ancora associato all'indirizzo IP corretto. Inoltre, sempre secondo quanto sopra espresso, potrete escludere qualsiasi comunicazione con un host reindirizzando il traffico a questi destinato verso la localhost. Ad esempio potrete applicare tale misura agli hostnames dei servers dedicati al servizio ads [ovvero all'invio dei banner pubblicitari], rendendo la navigazione più fluida e meno distraente dal momento che i loro annunci non potranno più essere caricati.

8.4. Database utenti e gruppi

L'elenco utenti si trova generalmente nel file `/etc/passwd`, mentre il file `/etc/shadow` contiene le password in hash. Entrambi sono file di testo, in un formato relativamente semplice, che possono essere letti e modificati con un editor di testo. Ogni utente è descritto su una riga con diversi campi separati dai due punti (":").

NOTA Come modificare i files di sistema	I files di sistema menzionati in questo capitolo sono in formato "testo puro" e possono quindi essere modificati con un editor di testo. Data la loro importanza per le funzionalità essenziali del sistema, vi raccomandiamo di adottare precauzioni extra prima di qualsiasi modifica ai files di sistema. Innanzitutto effettuate sempre il backup o una copia dei files di sistema prima del loro opening o alterazione. Inoltre sui servers o su macchine dove più persone possono potenzialmente accedere allo stesso file contemporaneamente prendete ulteriori misure per prevenire che il file venga corrotto. Per fare ciò, potrete utilizzare il comando `vipw` per modificare `/etc/passwd` o `vigr` per `/etc/group`. Questi comandi bloccheranno il file in questione in modo che non possa essere eseguito un editor di testo (`vi` per impostazione predefinita, a meno che non sia stata modificata la variabile di ambiente EDITOR). L'opzione `-s` di questi comandi consente di modificare il file `shadow` corrispondente.

BASILARE Crypt, una funzione a senso unico	`crypt`, è una "funzione unidirezionale" che trasforma una stringa (A) in un'altra stringa (B) in modo che da B non si possa risalire ad A. L'unico modo per identificare A è quello di testare tutti i potenziali valori corrispondenti, verificandone individualmente l'avvenuta trasformazione attraverso la funzione con il risultato concreto della stessa funzione ovvero B. Crypt utilizza fino a 8 caratteri come input (stringa A) e genera una stringa di 13 caratteri ASCII printable [tradotto letteralmente "stampabili"] (stringa B).

8.4.1. Elenco degli utenti: /etc/passwd

Questa è la lista dei campi del file /etc/passwd:
• login, ad esempio hertzog;
• password: è una password crittografata dalla funzione unidirezionale (crypt), che sfrutta DES, MD5, SHA-256 o SHA-512. Il valore speciale "x" indica che la password crittografata è conservata in /etc/shadow;
• uid: numero univoco che identifica ciascun utente;
• gid: numero univoco del gruppo principale dell'utente (Debian crea di default un gruppo specifico per ciascuno utente);
• GECOS: campo informativo che di solito contiene il nome completo dell'utente;
• login directory, assegnata all'utente per conservare i suoi files personali (la variabile d'ambiente `$HOME` di solito punta a questa directory);
• program to execute upon login. [programma da eseguire dopo il login.] È generalmente un interprete dei comandi (shell), che conferisce all'utente molta libertà. Se si specifica /bin/false (non viene eseguito alcun programma e non viene concesso alcun controllo), l'utente non sarà in grado di eseguire il login.

BASILARE Unix Group	Un gruppo Unix è un'entità [in breve "un'entità" in informatica è una `Primitive Data Type` ovvero non letteralmente un blocco di dati essenziale che comunica all'interprete dei comandi come utilizzare i dati] che include diversi utenti in modo che possano facilmente condividere files utilizzando l'integrated permission system (beneficiando degli stessi diritti). Potrete anche limitare l'uso di determinati programmi ad un determinato gruppo.

8.4.2. Il file /etc/shadow, criptato e nascosto

Il file /etc/shadow contiene i seguenti campi:
• login;
• encrypted password (password crittografata);
• diversi campi si occupano della scadenza della password.

SICUREZZA La protezione applicata al file /etc/shadow	/etc/shadow, a differenza della sua alternativa /etc/passwd, è inaccessibile alla lettura degli utenti regolari. Qualsiasi password crittografata conservata in /etc/passwd è leggibile da chiunque; quindi un cracker potrebbe provare a "rompere" la crittografia (e poi eventualmente rivelarla pubblicamente) attraverso uno dei diversi attacchi con metodo "forza bruta", che mettono semplicemente insieme le combinazioni di caratteri maggiormente utilizzate. Questo tipo di attacco è chiamato "a dizionario", ma non è più applicabile sui sistemi che adottano il file /etc/shadow.

DOCUMENTAZIONE I formati dei files /etc/passwd, /etc/shadow e /etc/group	Questi formati sono documentati nelle seguenti man pages: passwd(5), shadow(5), group(5).

8.4.3. Modifica un account o una password esistenti

I seguenti comandi consentono la modifica delle informazioni conservate nei correlati campi del database utente: `passwd` consente agli utenti regolari di cambiare la propria password, aggiornando in pratica il file `/etc/shadow`; `chfn` (CHange Full Name), riservato al super-user (root), modifica il campo GECOS; `chsh` (CHange SHell) consente agli utenti di cambiare la loro login shell, o shell, ma limitatamente fra quelle elencate nel file `/etc/shells`; diversamente l'amministratore non è limitato da questa restrizione e può impostare qualunque programma-shell desideri.
Infine, il comando `chage` (CHange AGE) consente all'amministratore di modificare i parametri della scadenza della password-utente (l'opzione `-l user` consente la visualizzazione della configurazione corrente). Potrete anche forzare la scadenza della password-utente usando il comando `passwd -e user`, che costringerà l'utente coinvolto dal suddetto comando a cambiare la sua password durante il successivo login.

8.4.4. Come bloccare un account

Potreste essere costretti a "bloccare l'account" di un utente, come misura disciplinare, nell'ambito di un'indagine o semplicemente nei casi di prolungato o permanente inutilizzo del suddetto account da parte dell'utente. L'effetto di un account disabilitato è che l'utente correlato non può più identificarsi attraverso il login o accedere alla macchina. L'account rimane intatto sulla macchina ed i suoi files o dati non vengono cancellati; è semplicemente non utilizzabile. Ciò è realizzabile attraverso il comando utente `passwd -l user` (da "lock" che significa in inglese bloccare). L'account può essere nuovamente abilitato attraverso un metodo simile, ovvero con l'opzione `-u` (da "unlock" che significa in inglese sbloccare).

ANDANDO OLTRE Database di sistema e NSS	Invece di utilizzare i soliti files per gestire gli elenchi di utenti e gruppi, potrete ricorrere ad altri tipi di database - come ad esempio LDAP o db - utilizzando un appropriato modulo NSS (Name Service Switch o service multiplexer). I moduli utilizzati sono elencati nel file /etc/nsswitch.conf sotto le voci `passwd`, `shadow` e `group`. Andate a leggere il paragrafo 11.7.3.1 “Come configurare l’NSS” a pagina 313 per avere un esempio concreto sull’uso del modulo NSS attraverso LDAP.

8.4.5. Elenco dei gruppi: /etc/group

L'elenco dei gruppi è memorizzato nel file `/etc/group`, un semplice database testuale in un formato simile a quello del file `/etc/passwd`, che utilizza i seguenti campi:

- group name (identificatore o il nome del gruppo);
- password (opzionale): viene utilizzata per "associarsi" ad un gruppo quando non si è membri abituali (attraverso il comando `newgrp` o `sg` - andate a leggere la casella di testo “Come lavorare con più gruppi” a pagina 175);
- gid: numero univoco che identifica il gruppo;
- list of members (l'elenco dei membri): l'elenco dei nomi degli utenti appartenenti al gruppo, separati da virgole.

BASILARE
Come lavorare con più gruppi

Qualsiasi utente può quindi far parte di più gruppi; fra questi vi è il “main group” (gruppo principale). Il “main group” predefinito dell’utente viene impostato durante la creazione dell’account utente.
Per impostazione predefinita, ogni file creato dall’utente appartiene ai gruppi correlati all’utente, nonché al main group. Ma non sempre questo è quello che si desidera: ad esempio, qualora lavoriate in una directory condivisa attraverso un altro gruppo diverso dal vostro main group. In questo caso, un utente potrebbe avere l’interesse di cambiare temporaneamente il main group attraverso uno dei seguenti comandi: `newgrp` - che avvia una nuova shell - o `sg` - che esegue semplicemente un comando utilizzando un gruppo alternativo sostituto. Questi comandi consentono inoltre ad un utente di unirsi ad un gruppo di cui non è membro. Se il gruppo è protetto da una password nota occorrerà fornirla prima di eseguire il comando.
In alternativa potete impostare il setgid bit sulla directory, in modo che i files creati in questa directory appartengano automaticamente al gruppo giusto. Per maggiori dettagli, andate a leggere la casella di testo “setgid directory e sticky bit” a pagina 214.
Il comando `id` consente di verificare in qualsiasi momento lo stato degli utenti, il loro identificativo personale (variabile `uid`), il loro corrente main group (variabile `gid`) e l’elenco dei gruppi di cui sono membri (variabile `groups`).

I comandi `addgroup` e `delgroup` consentono rispettivamente di creare ed eliminare un gruppo. Il comando `groupmod` modifica le informazioni di un gruppo (il suo gid o identificatore). Il comando `gpasswd group` cambia la password di un gruppo, mentre il comando `gpasswd -r group` la cancella.

SUGGERIMENTO
`getent`

Il comando getent (get entries) consulta, nella modalità standard, i databases di sistema utilizzando le funzioni della correlata libreria, che a turno chiamano i moduli NSS configurati nel file `/etc/nsswitch.conf` . Il suddetto comando richiede uno o due argomenti: il nome del database da consultare ed una potenziale parola chiave per la ricerca. Ad esempio, il comando `getent passwd rhertzog` restituisce informazioni riguardo al database utente “dell’utente rhertzog”.

8.5. Creazione degli accounts

Una delle prime azioni che un amministratore è costretto a compiere quando configura una macchina è la creazione di un account utente. Un account utente è tipicamente realizzato attraverso il comando `adduser`, che necessita sotto le vesti di argomento di un nome-utente per il nuovo utente da creare.
Il comando `adduser` pone alcune domande prima di creare l'account, ma il suo utilizzo è abbastanza semplice. Il file di configurazione `/etc/adduser.conf`, include tutte le impostazioni riguardo alla creazione dell'utente: potrete quindi utilizzare questo file per fornire automaticamente un "quota" per ciascun nuovo utente attraverso la creazione di un "modello utente" o per cambiare la posizione degli accounts utente; quest'ultima possibilità, raramente impiegata, si dimostra efficace per esempio quando c'è un vasto numero di utenti e si desidera distribuire i loro accounts su più dischi. Potrete anche scegliere una differente shell predefinita.

BASILARE
Quota

Il termine “quota” si riferisce ad una limitazione delle risorse della macchina che un utente può utilizzare. Questo limite spesso riguarda lo spazio sul disco.

La creazione dell'account crea la home directory dell'utente il cui contenuto deriva dal modello `/etc/skel/`. Così facendo l'utente viene provvisto delle directories e dei files standard.
In alcuni casi, sarà utile aggiungere un utente ad un gruppo (diverso dal suo "main group"), in particolare per concedergli ulteriori diritti. Ad esempio, un utente incluso nel gruppo audio sarà in grado di accedere ai dispositivi audio (andate a leggere la casella di testo "Diritti di accesso ad un dispositivo" a pagina 176). Per aggiungere un utente ad un gruppo eseguite il comando `adduser user group`.

BASILARE Diritti di accesso ad un dispositivo	Ogni dispositivo hardware è rappresentato nei sistemi Unix attraverso un cosiddetto “special files” [o file “device”], generalmente memorizzato nell’albero /dev/(DEVices). Esistono due tipi di special files a seconda della natura del dispositivo: i files in “character mode” ed i files in “block mode”; ciascuna modalità consente solo un numero limitato di operazioni. Mentre la “character mode” si limita alle interazioni con operazioni di lettura e scrittura, la “block mode” consente anche la ricerca nei dati disponibili. Infine, ogni special file è associato a due numeri (“major” e “minor”) che identificano per il kernel in modo univoco il dispositivo [il numero “major”, comune per tutti i dispositivi controllati dallo stesso driver, identifica per il kernel il tipo di dispositivo; il numero “minor” invece identifica per il driver le caratteristiche peculiari del dispositivo in modo da renderle accessibili]. Tale file, creato attraverso il comando `mknod`, ha semplicemente un nome simbolico (è più pratico per l’utente “umano”). I permessi di uno special file rilevano i permessi per accedere al dispositivo stesso. Pertanto, ad esempio, un file come /dev/mixer - ovvero il mixer audio - è accessibile in modalità lettura/scrittura solo all’utente root ed ai membri del gruppo `audio`. Di conseguenza solo questi utenti possono utilizzare il mixer audio. Si precisa però che la combinazione di `udev` e `policykit` è in grado di aggiungere ulteriori autorizzazioni per consentire agli utenti fisicamente connessi alla console (e non tramite la rete) di accedere ai diversi dispositivi.

8.6. Ambiente Shell

Gli interpreti dei comandi (o shells), essendo delle potenziali prime forme di contatto dell'utente con il computer, dovrebbero essere piuttosto intuitive. La maggior parte usano gli scripts di inizializzazione che consentono la configurazione del loro funzionamento (completamento automatico, prompt text, ecc.).
bash, la shell predefinita, usa lo script di inizializzazione /etc/bash.bashrc (per le "interactive shells") e /etc/profile (per i "login shells").

BASILARE Login shell e interactive shell (e non)	In poche parole, una login shell viene invocata quando accedete alla console tramite ssh o tramite il comando esplicito `bash --login`. Che si tratti di una login shell o meno, una shell può essere interattiva (se si svolge ad esempio in un terminale `xterm-type`); oppure una shell non interattiva (quando sta eseguendo uno script).

ESPLORANDO Altre shells, altri scripts	Ogni interprete di comandi ha una sintassi specifica e dei propri files di configurazione. Pertanto, `zsh` usa `/etc/zshrc` e `/etc/zshenv`; `csh` usa `/etc/csh.cshrc`, `/etc/csh.login` e `/etc/csh.logout` . Le man pages di questi programmi documentano i files che utilizzano.

Per bash, potrebbe esservi utile attivare il "completamento automatico" nel file `/etc/bash.bashrc` (semplicemente vi basterà decommentare poche righe).

BASILARE Completamento automatico	Molti interpreti di comandi supportano la funzionalità di completamento automatico che consente ad una shell di completare automaticamente il nome di un comando o l'argomento parzialmente inserito quando l'utente lo richiederà attraverso il tasto tab. In questo modo gli utenti potranno lavorare più velocemente e con meno rischi di errore. Questa funzionalità è molto efficace e flessibile. Difatti è possibile personalizzare il suo funzionamento per ciascun comando. Pertanto, il primo argomento configurato per il comando `apt` dovrà essere scelto in base alla sintassi dello stesso comando, anche se non lo relazionerete ad un file (in questo esempio le possibili scelte sono fra `install`, `remove`, `upgrade`, ecc.).

BASILARE La tilde, la scorciatoia per la HOME	La tilde viene spesso utilizzata per indicare la directory alla quale la variabile d'ambiente HOME punta (ovvero la home directory dell'utente, ad esempio `/home/rhertzog`). Pertanto gli interpreti dei comandi sostituiscono automaticamente `~/hello.txt` con `/home/rhertzog/hello.txt`. La tilde consente inoltre l'accesso alle home directory degli altri utenti. Di conseguenza, `~rmas/bonjour.txt` assume il significato di `/home/rmas/bonjour.txt`

Oltre ai suddetti scripts generici, ogni utente può creare i propri files `~/.bashrc` e `~/.bash_profile` per personalizzare la propria shell. Le modifiche più comuni sono la creazione di allases; gli aliases sono parole che vengono sostituite automaticamente durante l'esecuzione di un comando, il che velocizza l'invocazione dello stesso comando. Ad esempio, potrete generare l'alias di `la` per il comando `ls -la | less`; dopodiché vi basterà immettere `la` per ispezionare dettagliatamente i contenuti di una directory [senza dover immettere l'intero comando `ls -la | less`].

BASILARE Variabili d'ambiente	Le variabili di ambiente vengono utilizzate per memorizzare le global settings destinate alla shell o alle chiamate ai programmi. Sono contestuali (ogni processo ha il proprio set di variabili d'ambiente) ma ereditabili. Quest'ultima caratteristica offre l'opportunità ad una login shell di dichiarare le variabili che trasmetterà a tutti i programmi che eseguirà.

La configurazione delle variabili d'ambiente predefinite costituisce un elemento basilare della configurazione stessa della shell. Inoltre, non tenendo in considerazione le variabili peculiari ad una specifica shell, genericamente è preferibile posizionare le variabili nel file `/etc/environment`, dal momento che vengono impiegate dai vari programmi in grado di avviare una sessione shell. Tra le variabili tipicamente definite ci sono `ORGANIZATION`, che di solito comprende il nome dell'azienda o dell'organizzazione, e `HTTP_PROXY`, che dichiara l'esistenza e l'ubicazione di un proxy `HTTP`.

SUGGERIMENTO Configurare le shell in modo identico	Gli utenti spesso desiderano configurare le loro login ed interactive shells allo stesso modo. Per mettere in atto ciò, i suddetti utenti preferiscono interpretare (o usare come "sorgente") il contenuto del file `~/.bashrc` per il file `~/.bash_profile`. Potrete fare lo stesso con i files comuni a tutti gli utenti (attraverso la chiamata `/etc/bash.bashrc` da `/etc/profile/`).

8.7. Come configurare la stampante

La configurazione della stampante ha causato molti "mal di testa" sia ad amministratori, sia ad utenti. Per fortuna questi grattacapi sono ormai nella maggior parte dei casi un ricordo del passato, grazie a `cups`, un server di stampa gratuito che utilizza il protocollo IPP (Internet Printing Protocol).

Questo programma è suddiviso in diversi pacchetti Debian. La componente fondamentale del sistema è lo scheduler cupsd disponibile attraverso il pacchetto `cups-daemon`; `cups-client` include una serie di programmi per l'interazione con il server cupsd. lpadmin è probabilmente l'utility più importante anche se esistono degli strumenti per: abilitare o disabilitare una coda di stampa (printer queue); visualizzare o annullare la stampa in corso (print job); visualizzare o configurare le opzioni della stampante; ecc.. Il framework di CUPS si basa sul sistema di stampa di System V, ma attraverso il pacchetto `cups-bsd` potrete usare i comandi `lpr`, `lpq` e `lprm` che provengono dal sistema di stampa di BSD. [un compatibility layer, in informatica, è un'interfaccia che consente ai binari di sistemi legacy o foreign di poter essere eseguiti su un host in cui altrimenti sarebbero incompatibili]

CUPS COMUNITÀ	CUPS è un progetto ed un marchio registrato dell'azienda Apple. Prima dell'acquisizione da parte di Apple era denominato Common Unix Printing System ✦ https://www.cups.org/

Lo scheduler gestisce i print jobs per mezzo di un sistema di filtraggio allo scopo di produrre un file comprensibile per la stampante. Il sistema di filtraggio viene supportato dal pacchetto `cups-filters` (https://salsa.debian.org/printing-team/cups-filters/) in sinergia con i pacchetti `printer-driver-*`. Il sistema di stampa di Debian si basa sia su CUPS, sia sui pacchetti `cups-filters` e `printer-driver-*`.

Le stampanti prodotte e vendute negli ultimi 10 anni ormai sfruttano la tecnologia AirPrint, pertanto Debian Buster, CUPS e cups-filters hanno provveduto ad adeguare le loro funzionalità per questa nuova facility di rete. Inoltre le stampanti moderne sono stampanti IPP compatibili con il sistema di stampa driverless che consente il mero utilizzo di CUPS e di cups-filters. In questo modo non è più necessario installare il pacchetto printer-driver o il driver non-free del produttore per stampanti come Brother o Canon. Le stampanti USB possono beneficiare delle funzionalità delle stampanti moderne attraverso il pacchetto `ippusbxd`.

Il comando `apt install cups` installerà sia CUPS, sia `cups-filters`. Verrà anche installato il pacchetto raccomandato `printer-driver-gutenprint` compatibile con la maggior parte delle stampanti in circolazione, anche se potrebbe comunque essere necessaria l'installazione di un driver alternativo per alcuni dispositivi in particolare.

Per mezzo di `cups-browsed` le code di stampa e le moderne stampanti saranno rilevate via DNS Service Discovery - DNS-SD (Bonjour). Le stampanti USB dovranno essere configurate manualmente come descritto a seguire.

Dopo aver installato i summenzionati pacchetti, cups può essere amministrato facilmente grazie alla sua interfaccia web accessibile all'indirizzo locale: http://localhost:631. Sempre dalla sua interfaccia web potrete aggiungere, eliminare ed amministrare stampanti USB e network printers (stampanti di rete), gestendone le diverse funzionalità. Inoltre potrete amministrare `cups` anche attraverso l'interfaccia grafica fornita dall'ambiente desktop oppure con la GUI di `system-config-printer` (dall'omonimo pacchetto Debian).

8.8. Configurazione del Bootloader

Probabilmente il vostro boot loader è già funzionale, ma non fa male conoscere come configurarlo e installarlo qualora sparisse dal Master Boot Record. Ciò può accadere dopo l'installazione di un altro sistema operativo, come ad esempio Windows. La seguente informazione vi aiuterà anche a modificare la configurazione del boot loader se ne aveste bisogno.

BASILARE Master boot record	Il Master Boot Record (MBR) occupa i primi 512 byte del primo disco rigido ed è il primo elemento che viene caricato dal BIOS per conferire il controllo ad un programma in grado di effettuare il boot del sistema operativo desiderato. Generalmente, un bootloader viene installato sull'MBR rimuovendo il suo contenuto precedente

8.8.1. Come identificare i vostri dischi

CULTURA udev e /dev/	La directory /dev/ ospita tradizionalmente i cosiddetti "special" files, che rappresentano le periferiche del sistema (andate a leggere la casella di testo "Diritti di accesso ad un dispositivo" a pagina 176). In un lontano passato, era usata per contenere gli special files che potevano potenzialmente essere usati. Questo tipo di approccio presentava numerosi inconvenienti, tra i quali: il numero limitato dei dispositivi utilizzati (a causa dell'elenco hardcoded dei nomi) [Hard coding (o hard-coding o hardcoding) o codifica fissa (in italiano) è, in informatica e nello sviluppo software, la pratica di incorporare i dati direttamente nel codice sorgere di un programma o di un altro oggetto eseguibile, invece di ottenerli da sorgenti esterne o di generarli in fase di esecuzione]; l'impossibilità di conoscere quali special files erano davvero utilizzabili. Al giorno d'oggi, gli special files sono gestiti interamente in modo dinamico assecondando meglio la natura delle periferiche hot-swappable del computer. [In informatica un'interfaccia hot-swap, come ad esempio l'usb, consente il collegamento e/o lo scollegamento di un dispositivo hot-swappable anche a sistema avviato]. Il kernel collabora con udev (9.11.3, Come funziona udev - pag. 230) per creare o eliminare questi files quando i corrispondenti dispositivi appaiono o scompaiono. Per tale ragione non esiste più la necessità che /dev/ sia persistente e di conseguenza /dev/ è un RAM-based filesystem, all'avvio vuota e che contiene solo le voci pertinenti. Il kernel fornisce molte informazioni su un dispositivo quando viene aggiunto e rilascia una coppia di numeri (major/minor) per identificarlo. [Il numero "major", comune per tutti i dispositivi controllati dallo stesso driver, identifica per il kernel il tipo di dispositivo; il numero "minor" invece identifica per il driver le caratteristiche peculiari del dispositivo in modo da renderle accessibili.] udevd utilizza queste informazioni per creare uno special file, di cui stabilisce anche il nome e le relative autorizzazioni. Può anche creare aliases ed eseguire ulteriori azioni (tra cui attività di inizializzazione o di registrazione). Il funzionamento di udevd è regolamentato da un'ampia serie di regole (personalizzabili). Attraverso i nomi assegnati dinamicamente, potrete quindi mantenere lo stesso nome di un dato dispositivo, indipendentemente dall'ingresso/uscita a cui è collegato o, qualora fossero collegati più dispositivi (ad esempio più dispositivi USB), indipendentemente dall'ordine in cui i dispositivi sono stati collegati (rivelandosi molto utile). La prima partizione del primo disco è solitamente chiamata `/dev/sda1` per motivi di retro compatibilità, ma potrete decidere di chiamarla /dev/root-partition oppure di chiamarla in entrambi i modi contemporaneamente configurando udevd per creare automaticamente un symbolik link. In passato, alcuni kernel modules venivano caricati automaticamente quando si tentava di accedere al file del corrispondente device. Oggi è diverso e lo special file di una periferica non sarà presente se non viene prima caricato il modulo; il che non è molto grave in quanto la maggior parte dei moduli viene caricata all'avvio grazie al rilevamento automatico dell'hardware. Il rilevamento automatico dell'hardware però non funziona con le periferiche non rilevabili [undetectable peripherals] (come i vecchi disk drive o il mouse PS/2). Di conseguenza ricordate di aggiungere i moduli `floppy`, `psmouse` e `mousedev` in `/etc/modules` per forzarne il loro caricamento all'avvio.

La configurazione del bootloader deve consentirgli di identificare i diversi dischi e le loro partizioni. Per tale scopo Linux utilizza gli special files "block" conservati nella directory /dev/. A partire da Debian Squeeze, lo schema di denominazione degli hard drives è stato reso comune per tutti dallo stesso Kernel Linux e, di conseguenza, gli hard drives (IDE/PATA, SATA, SCSI, USB, IEEE 1394) sono ora rappresentati attraverso `/dev/sd*`.
Ciascuna partizione è quindi rappresentata da un numero in sequenza in base alla sua posizione sul disco: ad esempio `/dev/sda1` rappresenta la prima partizione sul primo disco, mentre `/dev/sdb3` rappresenta la terza partizione sul secondo disco.

L'architettura del PC (ovvero la "i386" e la sua cugina più giovane "amd64") è stata a lungo limitata all'utilizzo del formato "MS-DOS" della tabella delle partizioni, che consentiva solo quattro partizioni "primarie" per disco. Per superare questa limitazione, una delle suddette quattro partizioni viene realizzata come partizione "estesa", in modo che possa contenere partizioni "secondarie". Le partizioni secondarie vengono numerate con un numero maggiore o uguale a 5. Di conseguenza la prima partizione secondaria sarà la /dev/sda5, seguita dalla partizione /dev/sda6 e così via.

Un'altra limitazione della tabella delle partizioni MS-DOS è che non supporta dischi di dimensioni superiori a 2 TB, causando un notevole disagio con i dischi recenti.

GPT, un nuovo formato della tabella delle partizioni, consente di superare i summenzionati vincoli sul numero di partizioni (supporta fino a 128 partizioni) e sulla dimensione dei dischi (che può arrivare fino a 8 Zebibyte ZiB), che equivale a più di 8 bilioni di terabyte). Pertanto, se prevedete di creare più partizioni fisiche sullo stesso disco, dovrete creare una tabella delle partizioni in formato GPT durante la fase di partizionamento.

Purtroppo non è sempre facile ricordare quale disco avete collegato ad un dato controller SATA oppure alla terza posizione della SCSI chain, in quanto la denominazione degli hard drives hotplugged (ovvero la maggior parte dei dischi SATA e delle unità esterne) può cambiare da avvio ad avvio. Fortunatamente, udev crea, in aggiunta a /dev/sd*, dei symbolic links con nomi fissi, che se utilizzati possono identificare in maniera univoca gli hard drives. I suddetti symbolic links sono memorizzati in /dev/disk/by-id/. Ad esempio su una macchina con due dischi fisici, potrete trovare:

```
mirexpress:/dev/disk/by-id# ls -l
total 0
lrwxrwxrwx 1 root root 9 23 jul. 08:58 ata-STM3500418AS_9VM3L3KP -> ../../sda
lrwxrwxrwx 1 root root 10 23 jul. 08:58 ata-STM3500418AS_9VM3L3KP-part1
-> ../../sda1
lrwxrwxrwx 1 root root 10 23 jul. 08:58 ata-STM3500418AS_9VM3L3KP-part2
-> ../../sda2
[...]
lrwxrwxrwx 1 root root 9 23 jul. 08:58 ata-WDC_WD5001AALS-00L3B2_WD-WCAT00241697
-> ../../sdb
lrwxrwxrwx 1 root root 10 23 jul. 08:58 ata-WDC_WD5001AALS-00L3B2_WD-
WCAT00241697-part1 -> ../../sdb1
lrwxrwxrwx 1 root root 10 23 jul. 08:58 ata-WDC_WD5001AALS-00L3B2_WD-
WCAT00241697-part2 -> ../../sdb2
[...]
lrwxrwxrwx 1 root root  9 23 jul. 08:58 scsi-SATA_STM3500418AS_9VM3L3KP
-> ../../sda
lrwxrwxrwx 1 root  root 10 23 jul. 08:58 scsi-SATA_STM3500418AS_9VM3L3KP-part1
-> ../../sda1
```

```
lrwxrwxrwx 1 root root  10 23 jul. 08:58 scsi-SATA_STM3500418AS_9VM3L3KP-part2
-> ../../sda2
[...]
lrwxrwxrwx 1 root root  9 23 jul. 08:58 scsi-SATA_WDC_WD5001AALS-_WD-
WCAT00241697 -> ../../sdb
lrwxrwxrwx 1 root root  10 23 jul. 08:58 scsi-SATA_WDC_WD5001AALS-_WD-
WCAT00241697-part1 -> ../../sdb1
lrwxrwxrwx 1 root root  10 23 jul. 08:58 scsi-SATA_WDC_WD5001AALS-_WD-
WCAT00241697-part2 -> ../../sdb2
[...]
lrwxrwxrwx 1 root root 9 23 jul. 16:48 usb-LaCie_iamaKey_3ed00e26ccc11a-0:0 ->
../../sdc
lrwxrwxrwx 1 root root 10 23 jul. 16:48 usb-LaCie_iamaKey_3ed00e26ccc11a-0:0-
part1 -> ../../sdc1
lrwxrwxrwx 1 root root 10 23 jul. 16:48 usb-LaCie_iamaKey_3ed00e26ccc11a-0:0-
part2 -> ../../sdc2 [...]
lrwxrwxrwx 1 root root 9 23 jul. 08:58 wwn-0x5000c50015c4842f -> ../../sda
lrwxrwxrwx 1 root root 10 23 jul. 08:58 wwn-0x5000c50015c4842f-part1 -> ../../
sda1
[...]
mirexpress:/dev/disk/by-id#
```

Occorre precisare che alcuni dischi vengono elencati più volte (perché funzionano sia come dischi ATA, sia come dischi SCSI), ma le informazioni più rilevanti sono principalmente il modello ed il numero di serie dei dischi, attraverso cui potrete risalire al peripheral file.

I files di configurazione utilizzati come esempio nei successivi paragrafi sono tutti basati sulla medesima configurazione: un singolo disco SATA, in cui la prima partizione è dedicata ad una vecchia installazione di Windows, mentre la seconda è dedicata a Debian GNU/Linux.

8.8.2. Configurazione di LILO

LILO (Linux LOader) è il più vecchio bootloader — solido ma rustico. Il bootloader scrive sul MBR l'indirizzo fisico del kernel per il boot; ciò spiega perché ogni aggiornamento del kernel (o del file di configurazione di LILO) deve essere seguito dal comando `lilo`. Non dimenticatevene in quanto il sistema non sarà in grado di avviarsi se avrete rimosso o sostituito il vecchio kernel, in particolare perché il nuovo kernel non si troverà nella stessa posizione sul disco.
Il file di configurazione di LILO è `/etc/lilo.conf`; nell'esempio seguente troverete un semplice file con configurazione standard.

Esempio 8.4 File di configurazione LILO

```
# The disk on which LILO should be installed.
# By indicating the disk and not a partition.
# you order LILO to be installed on the MBR.
boot=/dev/sda
```

```
# the partition that contains Debian
root=/dev/sda2
# the item to be loaded by default
default=Linux


# the most recent kernel image
image=/vmlinuz
 label=Linux
 initrd=/initrd.img
 read only

Old kernel if the newly installed kernel doesn't boot)
image=/vmlinuz.old
  label=LinuxOLD
  initrd=/initrd.img.old
  read-only
  optional

# only for Linux/Windows dual boot
other=/dev/sda1
 label-Windows
```

8.8.3. La configurazione di GRUB 2

GRUB (GRand Unified Bootloader) è il più recente bootloader. Non dovrete invocarlo dopo ogni aggiornamento del kernel in quanto in grado di leggere i files system e di trovare autonomamente la posizione del kernel sul disco. Per installarlo sul MBR del primo disco, dovrete semplicemente digitare `grub-install /dev/sda`.

NOTA I nomi dei dischi con GRUB	GRUB è in grado di identificare gli hard drives grazie alle informazioni fornite dal BIOS. (hd0) corrisponde al primo disco rilevato, (hd1) al secondo, ecc. Nella maggior parte dei casi, questo ordine corrisponde esattamente al consueto ordine dei dischi in Linux, ma possono sorgere delle anomalie quando si utilizzano sia dischi SCSI, sia dischi IDE. GRUB salva le corrispondenze che trova nel file `/boot/grub/device.map`. Inoltre GRUB, onde evitare le anomalie quando genera `grub.cfg`, utilizza gli UUIDs o le labels (etichette) del file system. In ogni caso se l'ambiente corrente è diverso da quello di avvio potrete usare il file device map, dato che non è ancora obsoleto per override, (ignorare o ignorare e sostituire con...). Qualora troviate nel suddetto file degli errori (perché consapevoli che il BIOS rileva i dischi in un altro ordine) dovrete correggerli manualmente ed eseguire di nuovo `grub-install`. `grub-mkdevicemap` vi aiuterà a creare un file `device.map` da cui iniziare. Le partizioni in GRUB inoltre hanno un nome specifico. Se utilizzerete le partizioni "classiche" in formato MS-DOS, la prima partizione del primo disco sarà (hd0, msdos1), la seconda (hd0 msdos2) e così via.

La configurazione di GRUB 2 è conservata nel file `/boot/grub/grub.cfg`, ma questo file in Debian è generato da altri files. Fate quindi attenzione a non modificarlo manualmente, poiché perdereste le

modifiche locali apportate alla successiva esecuzione di `update-grub` (evento che potrebbe avverarsi durante l'aggiornamento dei diversi pacchetti). Le modifiche più comuni al file `/boot/grub/grub.cfg` (ad esempio per aggiungere dei parametri da riga di comando per il kernel o per cambiare il tempo di visualizzazione del menu) vengono realizzate attraverso le variabili definite in `/etc/default/grub`. Per aggiungere delle voci al menu, potrete creare un file `/boot/grub/custom.cfg` oppure modificare il file `/etc/grub.d/40_custom`. Per personalizzazioni più complesse, potrete aggiungere o modificare altri files in /etc/grub.d/; questi scripts dovrebbero restituirvi dei configuration snippets, possibilmente usufruendo di programmi esterni. [Nella programmazione gli "snippets" sono dei frammenti di codice.] I suddetti scripts aggiorneranno per primi l'elenco dei kernels per il boot: `10_linux` si occupa dei kernels Linux installati, `20_linux_xen` si occupa dei sistemi di virtualizzazione Xen e `30_os-prober` si occupa di altri sistemi operativi (Windows , OS X, Hurd).

8.9. Altre configurazioni: sincronizzazione dell'orologio, logs, condivisione ...

Questo paragrafo riunisce molti elementi utili per padroneggiare tutti gli aspetti della configurazione del sistema GNU/Linux. Tuttavia sono stati trattati brevemente e spesso rimandano alla documentazione di riferimento.

BASILARE Il collegamento simbolico	Un symbolic link è un pointer ad un altro file. [In informatica un pointer è un oggetto del linguaggio di programmazione che conserva un memory address, ovvero un riferimento ad una specifica posizione della memoria.] Quando utilizzate un pointer, viene aperto il file a cui “punta”. Inoltre la rimozione del suddetto collegamento non cancella il file “puntato”. Allo stesso tempo, non ha un proprio “set of permissions”, bensì adotta i permessi del target a cui punta. In conclusione, può puntare a qualsiasi tipo di file: directory, special file (socket, named pipes, device file, ecc.) o persino ad un altro symbolic link ... Il comando `ln -s target link-name` crea un symbolic link con un name-link che punta al target definito nello stesso comando. Se il target non esiste, il collegamento sarà “broken” [“broken” tradotto letteralmente in italiano significa “rotto” ed in questo caso è da intendersi come “non funzionante”] e se lo utilizzerete vi restituirà un errore che ribadirà l’inesistenza del file richiesto. Se il collegamento punta ad un altro collegamento otterrete una “chain” (“catena”) di collegamenti, che si trasformerà in un “cycle” (“ciclo”) se uno dei targets, a sua volta, punta ad uno dei collegamenti che lo hanno preceduto. In quest’ultimo caso, se utilizzerete uno dei collegamenti del cycle otterrete un errore specifico (“too many levels of symbolic links” ovvero “troppo livelli di collegamenti simbolici”), in quanto il kernel interromperà il cycle dopo averlo eseguito diverse volte.

8.9.1. Timezone — Fuso orario

Il fuso orario, configurato durante l'installazione iniziale, è un elemento di configurazione per il pacchetto `tzdata`. Per modificarlo, dovrete utilizzare il comando `dpkg-reconfigure tzdata`, che vi consentirà, di scegliere in modo interattivo il fuso orario. La sua configurazione è memorizzata nel file `/etc/timezone`. Inoltre, il file corrispondente della directory `/usr/share/zoneinfo/` viene copiato in `/etc/localtime`; questo file contiene nello specifico le regole che amministrano la modifica dell'ora per quei paesi che applicano l'ora legale.
Per modificare temporaneamente il fuso orario, potrete utilizzare la variabile d'ambiente `TZ`, che acquisisce una priorità rispetto alle impostazioni di sistema predefinite.

```
$ date
```

```
Thu Feb 19 11:25:18 CET 2015
$ TZ="Pacífic/Honolulu" date
Thu Feb 19 00:25:21 HST 2015
```

NOTA
Orologio di sistema, orologio hardware

In realtà ci sono due sorgenti per l'ora in un computer. La scheda madre del computer ha un orologio hardware, noto come "CMOS clock". Questo orologio non è molto preciso ed offre tempi di accesso piuttosto lenti. Il kernel del sistema operativo possiede a sua volta un proprio orologio, l'orologio di sistema, che aggiorna l'orario con i propri mezzi (possibilmente utilizzando dei time servers, andate a vedere il paragrafo 8.9.2, "Sincronizzazione dell'orologio" a pagina 184). L'orologio di sistema è generalmente più accurato, in particolare perché non necessita dell'accesso alle hardware variables. Tuttavia, poiché viene conservato solo nella live memory, viene azzerato ad ogni avvio, a differenza dell'orologio CMOS, che ha una batteria e che quindi "sopravvive" al riavvio o allo spegnimento della macchina.
L'orologio di sistema, all'avvio del computer, viene pertanto impostato in base all'orologio CMOS, e l'orologio CMOS, a sua volta, viene aggiornato con lo spegnimento ["shutdown"] (per salvare cambiamenti o correzioni qualora non fosse stato in precedenza regolato propriamente).
In pratica, il problema consiste nel fatto che l'orologio CMOS è solo un contatore e non contiene informazioni sul fuso orario. Dovrete quindi scegliere se "interpretarlo" come "tempo coordinato universale" o "tempo civile"(UTC, in precedenza GMT) oppure come "ora locale". La suddetta scelta dovrebbe essere semplice, ma di fatto le cose sono più complicate poiché il cambio dell'ora legale (Daylight Saving Time - DST) non è costante; l'effetto di ciò è che il sistema non ha modo di determinare se è corretto l'offset, in particolare nei periodi in cui il cambio di orario è prossimo. Pertanto vi consigliamo vivamente di impostare l'orologio CMOS con il "tempo coordinato universale" essendo tra l'altro sempre possibile risalire all'ora locale dall'ora e dal fuso orario universali.
Sfortunatamente, i sistemi Windows (nella loro configurazione predefinita) non consentono di applicare la summenzionata raccomandazione; mantengono l'orologio CMOS impostato con l'ora locale ed applicano il cambio dell'ora durante il boot del computer, prevedendo nei periodi in cui è prossimo il cambio dell'ora se già è in atto o meno. Questo sistema funziona relativamente bene fintanto sul computer è caricato solo Windows. Difatti non appena il computer utilizza diversi sistemi (sia in dual-boot o attraverso macchine virtuali), ne consegue il caos ed in nessun modo potrete avere la certezza che l'ora è corretta.
Se dovete necessariamente mantenere Windows su un computer, dovrete configurarlo in modo che il CMOS clock sia impostato come UTC (configurando il registry key `HKLM\SYSTEM\CurrentControlSet\Control\TimeZoneInformation\RealTimeIsUniversal` a "1" come `DWORD`), oppure utilizzando `hwclock --localtime --set` sul sistema Debian per configurare l'orologio hardware per il tracciamento dell'ora locale (ma verificate l'orologio hardware manualmente in primavera ed in autunno).
[Una dword, che è l'abbreviazione di "double word", è una typedef (una parola chiave dei sistemi di programmazione C e C++ utilizzata per assegnare dei nomi alternativi a dei tipi di dati esistenti) specifica dei sistemi Microsoft Windows.]

8.9.2. Sincronizzazione dell'orologio

La sincronizzazione dell'orologio, che può sembrare superflua su un computer, invece è davvero importante in una rete. Dal momento che gli utenti non hanno il diritto di modificare la data e l'ora, è importante che queste informazioni siano accurate in modo da scongiurare il caos. Inoltre, se tutti i computers di una rete sono sincronizzati il cross-referecing delle informazioni ottenute dai logs sulle differenti macchine sarà migliore. Di conseguenza, qualora avvenisse un attacco, sarà più semplice ricostruire la sequenza cronologica delle azioni malevoli sulle diverse macchine compromesse. Se le diverse macchine non sono sincronizzate i dati statistici raccolti sulle stesse macchine non hanno molto senso.

BASILARE NTP	NTP (Network Time Protocol) consente ad una macchina di sincronizzarsi con le altre in modo abbastanza preciso, tenendo conto dei ritardi indotti dal trasferimento delle informazioni sulla rete e da altri possibili offsets. [Molto genericamente l'offset è un numero intero che indica la distanza o lo slittamento dall'inizio di un object o di un array ad un loro dato elemento o ad una loro data posizione]. Sebbene ci siano molti servers NTP su Internet, i più popolari potrebbero essere sovraccarichi. Per questo motivo vi consigliamo di utilizzare il server NTP pool.ntp.org ossia un gruppo di macchine che funzionano come servers NTP pubblici. Potrete persino limitare il suddetto servizio ad un sotto-gruppo relativo ad un paese, ad esempio: us.pool.ntp.org per gli USA; ca.pool.ntp.org per il Canada; fr.pool.ntp.org per la Francia; ecc.. Tuttavia se gestite una vasta rete vi consigliamo di installare un server NTP personale che potrete sincronizzare con i servers pubblici. In questo caso, tutte le altre macchine della rete potranno usufruire del server NTP interno invece di incrementare il carico dei server pubblici. Potrete inoltre incrementare l'omogeneità degli orologi, visto che tutte le macchine saranno sincronizzate attraverso la stessa sorgente ... una sorgente molto vicina in termini di tempi di trasferimento nella rete.

8.9.2.1 Per le Workstations

Dato che le workstations sono riavviate regolarmente (anche per motivi di risparmio energetico), è sufficiente sincronizzarle tramite NTP all'avvio. Per ottenere ciò, installate semplicemente il pacchetto Debian `ntpdate`. Potete modificare il server NTP da utilizzare attraverso il file `/etc/default/ntpdate`.

8.9.2.2 Per i servers

I servers vengono riavviati molto raramente ed è molto importante che l'ora di sistema sia corretta. Per mantenere un orario corretto in modo permanente, dovrete installare un server NTP locale, grazie al servizio offerto dal pacchetto `ntp`. Nella sua configurazione predefinita, il server si sincronizzerà con pool.ntp.org e supporterà le macchine della rete locale che gli richiedono l'orario. Potrete configurarlo tramite il file /etc/ntp.conf, attraverso cui, cosa più importante, potrete cambiare il server NTP di riferimento. Se ci sono molti servers sulla stessa rete, potrebbe esservi utile possedere un solo un time server locale che si sincronizza con i server pubblici e che nel frattempo funge da sorgente agli altri servers della rete.

ANDANDO OLTRE Modulo GPS ed altre time sources	Se la sincronizzazione dell'orologio è particolarmente cruciale nella vostra rete, potete equipaggiarla di un server con un modulo GPS (che richiederà l'ora dai satelliti GPS) o un modulo DCF-77 (che sincronizzerà l'ora basandosi sull'orologio atomico installato vicino a Francoforte). In questi casi, la configurazione del server NTP è un po' più complicata ed è assolutamente necessaria la consultazione della documentazione.

8.9.3. Log Rotation

Man mano che i files di log crescono, è necessario archiviarli. Un modello comune è il "rotating archive": il file di log viene archiviato regolarmente e sono conservati solo i suoi ultimi archivi X.

logrotate, il programma responsabile per le "rotazioni", segue le direttive impartite nel file /etc/logrotate.conf e tutte quelle contenute nei files della directory /etc/logrotate.d/.
L'amministratore può modificare questi files se desidera personalizzare la policy del log rotation definita da Debian. La man page di logrotate(1) descrive tutte le opzioni consentite nei suddetti files di configurazione. Ad esempio potreste voler incrementare il numero di files conservati durante la log rotation oppure potreste desiderare di spostare i files di log in una directory specifica dedicata all'archiviazione anziché eliminarli. Potrete inoltre inviarli attraverso l'e-mail allo scopo di archiviarli altrove.
Il programma logrotate viene eseguito quotidianamente dal programma di scheduling (o scheduler) cron (descritto nel paragrafo 9.7, "Scheduling Tasks attraverso cron e atd" a pagina 221).

[In poche parole e molto genericamente lo scheduling è un metodo attraverso il quale un compito viene distribuito da un programma (scheduler) a delle risorse che lo mettono in atto e lo completano.]

8.9.4. Condivisione dei diritti di amministratore

Abbastanza spesso, diversi amministratori si prendono cura della stessa rete. La condivisione della password dell'utente root non è una soluzione molto elegante e potrebbe facilitare dei comportamenti scorretti a causa dell'anonimato di un account condiviso. La soluzione a questo problema è il programma sudo, che consente a determinati utenti di eseguire determinati comandi con "special rights" ["Special Rights" è anche un'espressione giuridica del common law che fa riferimento a leggi che concedono diritti a uno o più gruppi che non sono estesi ad altri gruppi. L'espressione giuridica di Special Rights assume una posizione per così dire controversa in quanto in contrasto con il principio di uguaglianza di tutti gli individui di fronte alla legge - può essere tradotto non letteralmente in questo caso come "privilegi speciali"]. Nella maggior parte degli use cases, sudo consente ad un utente fidato di eseguire qualsiasi comando come root. Per fare questo, l'utente deve semplicemente eseguire il comando sudo ed autenticarsi utilizzando la propria password personale.
Una volta installato, il pacchetto sudo fornisce diritti di root completi a tutti gli utenti membri del gruppo Unix sudo. Per delegare altri diritti, l'amministratore deve utilizzare il comando visudo, che consente di modificare il file di configurazione /etc/sudoers (che invocherà l'editor di testo vi o qualsiasi altro editor menzionato nella variabile d'ambiente EDITOR). Aggiungendo la riga username ALL= (ALL) ALL l'amministratore consentirà all'utente citato nella suddetta riga di eseguire qualsiasi comando come root.
Sono inoltre possibili configurazioni più sofisticate che consentono all'amministratore di condividere solo dei comandi specifici a determinati utenti. Tutti i dettagli delle diverse personalizzazioni sono trattati nella man page sudoers(5).

8.9.5. Elenco dei Mount Points

BASILARE Mounting e unmounting	In un sistema di tipo Unix come Debian, i files sono organizzati in una singola gerarchia a struttura ad albero di directories. La directory / si chiama “root directory”; tutte le altre directories sono sottodirectories contenute dalla root directory. “Mounting” è un’azione attraverso cui è possibile integrare il contenuto di una periferica (spesso un disco rigido) nella general file tree del sistema [che si distingue dal binary tree]. Pertanto, per poter utilizzare un disco esterno per memorizzare i dati personali degli utenti dovrà essere “montato” nella directory /home/. Il root filesystem [il “root filesystem” è il filesystem contenuto nella stessa partizione dove si trova la root directory ed è il file system sul quale tutti i filesystems sono “mounted”] è sempre montato dal kernel all’avvio del computer; altri dispositivi sono spesso montati dopo l’avvio, durante la sequenza di startup oppure manualmente attraverso il comando mount. Alcuni dispositivi rimovibili vengono montati automaticamente dopo essere stati collegati, specialmente se si utilizzano gli ambienti desktop GNOME e Plasma. Altri devono essere montati necessariamente manualmente dall’utente. Inoltre quest’ultimi dispositivi devono essere anche smontati (o rimossi dal file tree) manualmente. Agli utenti di basso profilo non è consentito utilizzare i comandi mount ed unmount. L’amministratore può tuttavia autorizzare queste operazioni (indipendentemente per ciascun mount point) configurando l’opzione user nel file /etc/fstab.

Il comando `mount` può essere utilizzato senza argomenti per elencare tutti i filesystems montati; per visualizzare i files system da fstab dovrete eseguire `findmnt --fstab`. I seguenti parametri sono richiesti per il montaggio o lo smontaggio di un device. Per maggiori informazioni fate riferimento alle correlate man pages, `mount(8)` e `unmount(8)`. Per usi basilari, la sintassi è semplice: per esempio, per montare la partizione `/dev/sdc1`, il cui filesystem è un ext3, nella directory /mnt/tmp/, dovrete solamente digitare `mount -t ext3 /dev/sdc1 /mnt/tmp/` .

Il file /etc /fstab fornisce l'elenco di tutti i possibili mounts eseguiti automaticamente all'avvio o da eseguire manualmente per i dispositivi di archiviazione rimovibili. Ogni mount point è descritto da una riga con diversi fields (space-separated):

• file system: definisce la posizione in cui è possibile trovare il file system da montare, può essere una partizione locale (un disco rigido, un CD-Rom) oppure un file system remoto (NFS).

Il suddetto field è spesso sostituito dall'identificatore univoco ID del filesystem (che può essere ottenuto con `blkid` **`device`**) con il prefisso UUID =. Ciò eviterà la modifica del nome del dispositivo qualora aggiungiate o rimuoviate dei dischi o qualora i dischi vengano rilevati in un ordine diverso.

• mount point (punto di mount): questa è la posizione nel filesystem locale in cui il dispositivo, il sistema remoto o la partizione saranno montati;
• type: questo field definisce il filesystem utilizzato sul dispositivo montato. ext4, ext3, vfat, ntfs, btrfs, xfs sono solo alcuni esempi.

BASILARE
NFS, un file system di rete

NFS - Network File System - è un network filesystem; sotto Linux, consente un “transparent access” ai files remoti integrandoli nel filesystem locale. [Per `transparent access` si intende che a prescindere dalla modalità di accesso ogni singolo utente potrà usufruire delle risorse in modo univoco ed uniforme].

L'elenco completo dei filesystems riconosciuti è disponibile nella man page mount (8). Lo special value `swap` indica le partizioni swap; lo special value `auto` dichiara al programma mount di rilevare automaticamente il filesystem (ciò è particolarmente utile sia per i lettori di dischi, sia per le chiavette USB, in quanto ciascuno di questi dispositivi potrebbe ospitare un diverso filesystem);

• options: sono numerose, dipendono dal filesystem e sono documentate nella man page mount. Le più comuni sono:

- `rw` o `ro` rispettivamente indicano che il filesystem sarà montato con permessi di read/write (lettura/scrittura) o con permessi read-only (solo lettura);

- `noauto` disabilita il mount automatico all'avvio;

- `nofail` indica al boot di procedere anche se il dispositivo non è presente. Non dovete dimenticare questa opzione per i dischi esterni che possono essere disconnessi all'avvio, altrimenti systemd interromperà il processo di boot fino a quando tutti i mount points da montare automaticamente non vengono effettivamente montati. Si precisa che potete coniugare la sopra menzionata opzione con `x-systemd.device-timeout=5s` per trasmettere a `systemd` di non attendere per più di 5 secondi che il dispositivo sia connesso (andate a leggere `systemd.mount(5)`);

- `user` autorizza tutti gli utenti a montare questo filesystem (operazione solitamente riservata a root);

- `defaults` indica l'insieme di opzioni predefinite (`rw`, `suid`, `dev`, `exec`, `auto`, `nouser` e `async`), che possono essere disabilitate individualmente aggiungendo dopo defaults le opzioni `nosuid`, `no-dev`, ecc. per disabilitare rispettivamente `suid`, `dev`, ecc. e così via ... L'opzione `user` abilita nuovamente i defaults (in quanto l'opzione defaults include anche `nouser`).

• dump (cattura): questo field è quasi sempre impostato a 0. Quando è impostato a `1`, dichiara al tool `dump` che la partizione contiene i dati di cui eseguire il backup.

• pass: quest'ultimo field indica se deve essere verificata l'integrità del filesystem all'avvio e in quale ordine di successione dovrebbe essere eseguita tale verifica. Se è impostato a 0, non viene eseguita alcuna verifica. Il root filesystem dovrebbe essere configurato con un valore pari ad `1`, mentre gli altri filesystem permanenti dovrebbero essere impostati con un valore pari a 2.

Esempio 8.5 Esempio di un file /etc/fstab

```
# /etc/fstab: static file system information.
#
# <file system>          <mount point>     <type>     <options>             <dump>  <pass>
proc                      /proc             proc       defaults                0       0
# / was on /dev/sda1 during installation
UUID=c964222e-6af1-4985-be04-19d7c764d0a7     / ext3 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=ee880013-0f63-4251-b5c6-b771f53bd90e none swap sw                        0       0
/dev/scd0                 /media/cdrom0   udf,iso9660  user,noauto             0       0
/dev/fd0                  /media/floppy   auto       rw,user,noauto           0       0
arrakis:/shared           /shared         nfs        defaults                 0       0
```

L'ultima voce in questo esempio corrisponde ad un fllesystem di rete (NFS): la directory `/shared/` del server `arrakis` è montata sulla directory `/shared/` della macchina locale. Il formato del file `/etc/fstab` è documentato nella man page `fstab(5)`

ANDANDO OLTRE Auto-mounting	`systemd` gestisce gli automount points, ossia i file system montanti su richiesta quando un utente tenta di accedere al loro “target” mount point. E smonterà tali dispositivi automaticamente se nessun processo vi sta accedendo. Alla stessa stregua di molti concetti di systemd, gli automount points sono gestiti attraverso delle units dedicate (attraverso il suffisso `.automount`) Per maggiori informazioni consultate `systemd.automount(5)`. Esistono altre utilities auto-mounting, ad esempio `automount` del pacchetto `autofs` o `amd` del pacchetto `am-utils`. Si precisa inoltre che gli ambienti desktop `GNOME`, Plasma, ecc. utilizzano `udisks` e possono montare automaticamente i dispositivi rimovibili quando vengono connessi.

8.9.6. locate e updatedb

Il comando `locate` può trovare la posizione di un file del quale conoscete solo parzialmente il nome. Restituisce un risultato quasi istantaneamente perché consulta un particolare database che memorizza la posizione di tutti i files nel sistema; questo database viene aggiornato quotidianamente dal comando `updatedb`. Esistono diverse implementazioni del comando `locate` e Debian ha scelto `mlocate` per il suo sistema standard.
`mlocate` è abbastanza veloce e, anche se si serve di un database che gli consente di conoscere di tutti i files sul sistema (in quanto la sua implementazione `updatedb` viene eseguita con autorizzazioni root), restituisce all'utente che avvia il comando solo i files che sono accessibili all'utente stesso. Per ragioni di sicurezza, l'amministratore può escludere determinate directories dall'indicizzazione, utilizzando la variabile `PRUNEDPATHS` nel file di configurazione `/etc/updatedb.conf`.

8.10. Come compilare un kernel

I kernels supportati da Debian includono il maggior numero possibile di funzionalità, così come la maggior parte di drivers, allo scopo di comprendere la più ampia gamma di configurazioni hardware esistenti. Per questo alcuni utenti preferiscono ricompilare il kernel per includere solo gli elementi di cui necessitano veramente. Ci sono due ragioni per una decisione simile. Innanzitutto, potrebbe essere attuata per ottimizzare il consumo di memoria, in quanto il codice kernel, anche se non viene mai utilizzato, occupa memoria per nulla (e non si trasferirà mai nella swap, perché utilizza l'effettiva RAM), il che potrebbe ridurre le prestazioni complessive del sistema. Difatti un kernel compilato localmente può limitare il rischio di problemi di sicurezza in quanto il codice compilato e che verrà processato è una frazione minore del codice kernel standard.

NOTA Aggiornamenti di sicurezza	Se scegliete di compilare un kernel, dovrete accettarne le conseguenze: Debian non fornirà aggiornamenti di sicurezza per il kernel che avete personalizzato. Invece utilizzando un kernel supportato da Debian, beneficerete degli aggiornamenti preparati dal team Debian Project's security.

La ricompilazione del kernel è anche necessaria se si desidera utilizzare diverse funzionalità che sono solo disponibili sotto forma di patches (e non sono incluse nella versione standard del kernel).

ANDANDO OLTRE Debian Kernel Handbook	I teams Debian Kernel gestiscono il "Debian Kernel Handbook" (disponibile nel pacchetto debian `debian-kernel-handbook`), che contiene una vasta documentazione sulla maggior parte delle kernel tasks e su come vengono gestiti i Debian kernel packages. Il "Debian Kernel Handbook" è il primo posto dove cercare se avete bisogno di più informazioni rispetto a quelle trattate da questo capitolo. ✦ https://kernel-team.pages.net/kernel-handbook/

8.10.1. Introduzione e Prerequisiti

Debian notoriamente gestisce il kernel sotto forma di pacchetto, pur non essendo un modo "tradizionale" per compilare ed installare un kernel. Dal momento che il kernel è sotto il controllo del

packaging system, può essere effettuata una sua rimozione pulita oppure può essere "deployed" su diverse macchine [ovvero può essere installato e configurato su diverse macchine e nel frattempo addestrato il personale al suo utilizzo durante la sua stessa installazione]. Inoltre, gli scripts associati a questi pacchetti automatizzano l'interazione con il bootloader e l'initrd generator. [In informatica initrd (initial ramdisk) è uno schema per caricare nella ram un temporary root file system, utilizzato come parte del processo di avvio di Linux.]
Le sorgenti upstream del kernel Linux contengono tutto quello che è necessario per compilare un pacchetto Debian del kernel. Dovrete anche installare il pacchetto `build-essential`, per essere certi di avere gli strumenti richiesti per generare un pacchetto Debian. Inoltre, la configurazione del kernel vi richiederà il pacchetto `libncurses5-dev`. Infine, il pacchetto `fakeroot` creerà il pacchetto Debian senza usare i diritti di amministratore.

CULTURA
I bei vecchi tempi del kernel-package

Prima che il Linux build system fosse capace di generare un pacchetto Debian il metodo raccomandato per creare questi pacchetti era utilizzare il tool `make-kpkg` del pacchetto `kernel-package`.

8.10.2. Come recuperare le Sorgenti

Anche le sorgenti del kernel Linux sono disponibili sotto forma di pacchetto, come del resto altre utilità del sistema Debian. Per recuperare le sorgenti, dovrete installare soltanto il pacchetto `linux-source-version`. Attraverso il comando `apt search ^linux-source` otterrete l'elenco delle diverse versioni del kernel "packaged" ("impacchettate") da Debian. Le versioni più recenti sono disponibili nella distribuzione Unstable: potrete recuperale senza grandi rischi (specialmente se APT è stato configurato seguendo le istruzioni del paragrafo 6.2.6, "Come utilizzare più distribuzioni" a pagina 124). Si precisa che i codici sorgente contenuti in questi pacchetti non corrispondono esattamente a quelli pubblicati da Linus Torvalds e dagli sviluppatori del kernel; come tutte le distribuzioni anche Debian applica un certo numero di patches che, non è detto, durante il loro iter, debbano trovarsi nelle versioni upstream di Linux. Queste modifiche includono: backports di fixes/features/drivers dalla versione più recente del kernel Linux; nuove funzionalità non ancora integrate nella upstream Linux tree; nonché alcune funzionalità specifiche di Debian.

La parte successiva di questo capitolo sarà focalizzata sulla versione del kernel Linux 4.19, ma ovviamente potrete adattare gli esempi alla specifica versione del kernel che desiderate.

Presumiamo che abbiate installato il pacchetto `linux-source-4.19`. Questo pacchetto contiene il file `/usr/src/linux-source-4.19.tar.xz`, un archivio compresso di sorgenti del kernel. Dovrete decomprimere questi files in una nuova directory (e non direttamente in /usr/src, in quanto non sono necessari i permessi speciali per compilare un kernel Linux): `~/kernel/` sarà adatta.

```
$ mkdir ~/kernel; cd ~/kernel
$ tar -xaf /usr/src/linux-source-4.19.tar.xz
```

CULTURA
Posizione delle sorgenti del kernel

Tradizionalmente, le sorgenti del kernel Linux dovrebbero essere collocate in `/usr/src/linux`, con conseguenti permessi di root per la compilazione. Tuttavia, come sapete, dovreste evitare di utilizzare i diritti di amministratore, se non è strettamente necessario. In teoria il gruppo `src` consente ai suoi membri di utilizzare la directory `/usr/src/`, ma dovreste evitare comunque di farlo. Mantenendo le sorgenti del kernel in una directory personale, ne guadagnerete in sicurezza sotto ogni punto di vista: nessuno dei files `/usr/` sarà sconosciuto al packaging system e non rischierete di fuorviare i programmi che scansionano `/usr/src/linux` per ottenere informazioni sul kernel utilizzato.

8.10.3. Configurazione del kernel

Il prossimo passo che dovrete compiere è configurare il kernel in base alle vostre esigenze. L'esatta procedura da eseguire dipende dai traguardi che intendete raggiungere.
È probabile che durante la ricompilazione di una versione più recente del kernel (possibilmente con una patch aggiuntiva) scegliate una configurazione quanto più possibile vicina a quella standard offerta da Debian. In questo caso, invece di riconfigurare tutto da zero, dovrete semplicemente copiare il file `/boot/config-version` (ovvero la versione del kernel attualmente in uso, riscontrabile attraverso il comando `uname -r`) in un file .config nella directory delle sorgenti del kernel:

```
$ cp /boot/config-4.19.0-5-amd64 ~/kernel/linux-source-4.19/.config
```

Se non desiderate modificare la soprastante configurazione, potete fermarvi qui e passare direttamente al paragrafo 8.10.4, "Compiling [letteralmente "compilazione"] e Building [letteralmente "costruzione", "generazione"] di un pacchetto" a pagina 192. Altrimenti se avete deciso di riconfigurare tutto da zero, dovrete soffermarvi a configurare il vostro kernel.

Esistono diverse interfacce dedicate, nella directory "kernel source" [letteralmente "sorgente del Kernel"], che possono essere chiamate utilizzando il comando `make target`, dove "target" [letteralmente "bersaglio"] è uno dei valori descritti in seguito.

`make menuconfig` compila ed esegue un'interfaccia in modalità testuale (ed è necessario il pacchetto `libncurses5-dev`) che consente di "navigare" fra le opzioni supportate in una struttura gerarchica. Premendo il tasto `Space` modificherete il valore dell'opzione selezionata, mentre premendo il tasto `Enter` convaliderete il pulsante selezionato nella parte inferiore dello schermo: `Select` vi restituirà il sottomenu selezionato; `Exit` chiuderà la schermata corrente e ritornerà indietro ripercorrendo a ritroso la gerarchia; `Help` vi mostrerà delle informazioni più dettagliate riguardo al ruolo dell'opzione selezionata. I `tasti frecce` vi consentiranno di spostarvi tra l'elenco di opzioni e pulsanti. Per uscire dal programma di configurazione dovrete selezionare `Exit` dal menu principale. Dopodiché il programma vi proporrà di salvare le modifiche: accettate solo se vi riterrete soddisfatti delle scelte effettuate.

Le altre interfacce hanno funzionalità simili, ma hanno interfacce grafiche più moderne: `make xconfig` utilizza l'interfaccia grafica `Qt` e `make gconfig` utilizza `GTK+`. La prima necessita di `libqt4-dev` mentre la seconda dipende da `libglade2-dev` e `libgtk2.0-dev`.

Quando si utilizza una di queste interfacce di configurazione, è generalmente consigliabile iniziare da una configurazione predefinita ragionevole. Il kernel fornisce tali configurazioni in `arch/architecture/configs/*_defconfig` ed è possibile impostarle con un comando come `make x86_64_defconfig` (per un PC a 64 bit) o `make_i386_defconfig` (per un PC a 32 bit).

SUGGERIMENTO Cosa fare con un file .config obsoleto?	Se utilizzerete un file `.config` generato da un'altra versione del kernel (più vecchia), dovrete aggiornare quel file. Potrete farlo attraverso `make oldconfig`, che vi porrà in modalità interattiva delle domande sulle nuove opzioni di configurazione. Per utilizzare le risposte predefinite a tutte le suddette domande, potrete usare `make olddefconfig`. Infine, il comando `make oldnoconfig` farà in modo che le risposte a tutte le suddette domande siano negative.

8.10.4. Compiling [letteralmente "compilazione"] e Building [letteralmente "costruzione", "generazione"] di un pacchetto

NOTA Rimozione pulita prima di un rebuilding	Se avete già effettuato una compilazione del kernel nella directory e desiderate ricominciare tutto da capo (ad esempio perché avete modificato la configurazione del kernel), dovrete eseguire `make clean`, che eliminerà i files compilati. `make distclean` eseguirà una pulizia ancora più accurata ed eliminerà tutti i files generati, incluso il vostro file `.config`, pertanto fate prima un backup. Se intendete copiare il file di configurazione da `/boot/` dovrete anche modificare l'opzione del system trusted keys semplicemente immettendo una stringa vuota: `CONFIG_SYSTEM_TRUSTED_KEYS = “”`.

[Il termine inglese “Building” ed il suo derivato “Rebuilding” vengono utilizzati, anche in questo manuale, con il significato rispettivamente di “compilare” e di “ricompilare”]

Quando la configurazione del kernel sarà pronta, il comando `make deb-pkg` creerà fino a 5 pacchetti Debian: `linux-image-version`, che contiene l'immagine del kernel ed i moduli correlati; `linux-headers-version`, che contiene i files header necessari per compilare i moduli esterni; `linux-firmware-image-version`, che contiene i files del firmware richiesti da alcuni drivers (questo pacchetto potrebbe non essere incluso se si genera il kernel dalle sorgenti offerte da Debian); `linux-image-version-dbg`, che contiene i `debugging symbols` per la kernel image e per i suoi moduli, e `linux-libc-dev`, che contiene gli headers specifici di alcune librerie user-space, come ad esempio la libreria `GNU glibc`.
La versione viene definita attraverso il concatenamento della versione upstream (ovvero viene definita attraverso le variabili `VERSION`, `PATCHLEVEL`, `SUBLEVEL` ed `EXTRAVERSION` nel file `Makefile`), dal parametro di configurazione `LOCALVERSION` e dalla variabile d'ambiente `LOCALVERSION`. [Un `makefile` è un file attraverso cui viene gestito il frazionamento del programma `make`] La versione del pacchetto riutilizza la stessa "`version string`" con una revisione che viene regolarmente incrementata (e conservata nel file `.version`), a meno che non ne eseguiate l'override attraverso la variabile di ambiente `KDEB_PKGVERSION`.

```
$ make deb-pkg LOCALVERSION=-falcot KDEB_PKGVERSION=$(make kernelversion) -1
[...]
$ ls ../*.deb
../linux-headers-4.19.37-falcot_4.19.37-1_amd64.deb
../linux-image-4.19.37-falcot_4.19.37-1_amd64.deb
../linux-libc-dev_4.19.37-1_amd64.deb
```

8.10.5. Compilazione dei moduli esterni

Alcuni moduli sono manutenuti al di fuori del kernel Linux ufficiale. Per usarli dovrete compilarli insieme al kernel corrispondente. Una gran quantità di moduli di terzi sono offerti da Debian in pacchetti dedicati, come `vpb-driver-source` (moduli aggiuntivi per `Voicetronix telephony hardware`) o `leds-alix-source` (ovvero il driver per le schede `PCEngines ALIX 2/3`).

Esistono diversi pacchetti e persino variegati fra loro; il comando `apt-cache rdepends module-assistans$` vi consentirà di consultare quelli disponibili in Debian. In ogni caso, questo elenco non sarà particolarmente utile, dal momento che non esiste un motivo particolare per compilare i moduli esterni, tranne che siate a conoscenza di averne bisogno. Nel qual caso la documentazione del dispositivo vi specificherà dettagliatamente i moduli necessari per funzionare con Linux.

Ad esempio date un'occhiata al pacchetto `dahdi-source`: dopo la sua installazione, un file `.tar.bz2` delle sorgenti del modulo viene archiviato in `/usr/src/`. In realtà potreste già estrarre il tarball e generare il modulo, ma è consuetudine automatizzare il tutto con DKMS. La maggior parte dei moduli offre la necessaria integrazione con DKMS attraverso un pacchetto il cui nome termina con il suffisso `-dkms`. Riprendendo il soprastante esempio, vi basterà installare il pacchetto `dahdi-dkms` per compilare il kernel module per il kernel corrente, a condizione che abbiate installato anche il pacchetto `linux-headers-*` idoneo al kernel corrente. Per esempio, per usare `linux-image-amd64`, dovrete installare anche `linux-headers-amd64`.

```
$ sudo apt install dahdi-dkms

[...]
Setting up xtables-addons-dkms (2.12-0.1) ...
Loading new xtables-addons-2.12 DKMS files...
Building for 4.19.0-5-amd64
Building initial module for 4.19.0-5-amd64
Done.

dahdi_dummy.ko:
Running module version sanity check.
 - Original module
   - No original module exists within this kernel
 - Installation
   - Installing to /lib/modules/4.19.0-5-amd64/updates/dkms/
[...]
DKMS: install completed.
$ sudo dkms status
dahdi, DEB_VERSION, 4.19.0-5-amd64, x86_64: installed
$ sudo modinfo dahdi_dummy
filename: /lib/modules/4.19.0-5-amd64/updates/dkms/dahdi_dummy.ko
license: GPL v2
author: Robert Pleh <robert.pleh@hermes.si>
description: Timing-Only Driver
[...]
```

ALTERNATIVA
module-assistant

Prima della comparsa di DKMS, il `module-assistant` era la soluzione più semplice per la creazione ed il deployment dei kernel modules.
Questa soluzione è ancora utilizzata, in particolare per i pacchetti che non offrono ancora l'integrazione con DKMS: con un semplice comando come ad esempio `module-assistant auto-install dadhi` (o la versione abbreviata `m-a a-i dadhi`), i moduli saranno compilati per il kernel corrente, posti in un nuovo pacchetto Debian, che a sua volta sarà installato al volo.

8.10.6. Come applicare una patch del kernel

Alcune funzionalità non sono incluse nel kernel standard in quanto non ancora mature o per un disaccordo fra i manutentori del kernel. Di conseguenza alcune funzionalità potrebbero essere rilasciate sotto forma di patch, in modo che chiunque possa liberamente applicarle alle sorgenti del kernel.

Debian distribuisce alcune di queste patches attraverso i pacchetti `linux-patch-*` ma non tutte si trovano nei rilasci stable (a volte per le stesse ragioni per cui non sono state incluse nel kernel upstream ufficiale).
Questi pacchetti installano i files nella directory `/usr/src/kernel-patches/`.
Per applicare una o più patches installate, dovrete utilizzare il comando `patch` nella directory sorgente e poi avviare la compilazione del kernel come descritto precedentemente.

```
$ cd ~/kernelflinux-source-4.19
$ make clean
$ zcat /usr/src/kernel-patches/diffs/grsecurity2/
grsecurity-3.1-4.9.11-201702181444.patch.gz | patch -p1
```

Si precisa che una patch potrebbe non funzionare necessariamente con tutte le versioni dei kernel; è quindi possibile che la patch non riesca ad "applicarsi" alle sorgenti del kernel. Vi apparirà quindi un messaggio di errore che vi darà dei dettagli in merito: in questo caso, fate riferimento alla documentazione disponibile nel pacchetto Debian della patch (nella directory `/usr/share/doc/linux-patch-*/`). È probabile che il manutentore specifichi a quali versioni del kernel la patch era destinata.

8.11. Installazione di un kernel

8.11.1. Funzionalità di un Debian Kernel Package

Un Debian Kernel Package installa la kernel image (`vmlinuz-version`), la sua configurazione (`config-version`) e la sua symbols table (`System.map-version`) in `/boot/`. I moduli sono installati nella directory `/lib/modules/version/`.

CULTURA
La symbols table

La symbols table consente agli sviluppatori di comprendere il significato di un messaggio di errore del kernel; in assenza di una "`symbols table`" i "`kernel oopses`" - (un "`oops`" è l'equivalente kernel di un "`segmentation fault`" dei programmi dell'user-space, in altre parole questi messaggi sono conseguenti ad una dereferenziazione di un puntatore "non valido") - conterrebbero solo indirizzi di memoria numerici, che rappresenterebbero un'informazione inutile senza una "tabella" che si occupa del mapping tra questi indirizzi ed i symbols ed i function names.

[Molto genericamente: il "`kernel oop`" è un errore dovuto al funzionamento del kernel non appropriato, poi riportato come "error log" — il famoso "Kernel Panic" è la conseguenza a molti "kernel oopses" —; un "`segmentation fault`" o "errore di segmentazione" è un errore dovuto al fatto che un programma tenta di accedere, senza i relativi e dovuti permessi, ad una posizione della memoria.]

Gli scripts di configurazione del pacchetto generano automaticamente un'`initrd image`, che è un mini-system designato per essere caricato in memoria (da qui il nome "`init ramdisk`") dal bootloader ed usato dal kernel Linux al solo scopo di caricare i moduli necessari per accedere ai dispositivi contenenti il sistema Debian completo (ad esempio il driver per i dischi SATA). Infine, gli scripts post-installazione aggiornano i "symbolic links" (i collegamenti simbolici) `/vmlinuz`, `/vmlinuz.old`, `/initrd.img` e `/initrd.img.old` in modo che puntino rispettivamente agli ultimi due kernel installati ed alle loro corrispondenti immagini initrd.

La maggior parte di queste attività sono delegate agli "`hook scripts`" contenuti nelle directories `/etc/kernel/*.d/`. [Molto genericamente gli "hook scripts" sono dei programmi che si attivano a seguito di un repository event]. Ad esempio, l'integrazione attraverso grub si basa su `/etc/kernel/postinst.d/zz-update-grub` e su `/etc/kernel/postrm.d/zz-update-grub`, per chiamare `update-grub` quando i kernels vengono installati o rimossi.

8.11.2. Installazione con dpkg

Usare `apt` è così conveniente da far dimenticare facilmente i tools di più basso livello, tuttavia il modo più semplice per installare un compiled kernel è usare il comando `dpkg -i package.deb` dove package.deb rappresenta ovviamente il nome di un pacchetto `linux-image` come ad esempio `linux-image-4.19.37-falcot_1_amd64.deb`.

Le fasi di configurazione descritte in questo capitolo sono basilari tanto da poter essere destinate ad un server nonché ad una workstation, e replicate massivamente in modo semi-automatico. Tuttavia, tali fasi non sono sufficienti di per sé a garantire un fully configured system. Difatti occorre ancora configurare altro, a partire dai programmi di basso livello chiamati "Unix services" ("servizi Unix").

Parole chiave

Avvio del sistema
Initscripts
SSH
Telnet
Diritti (Rights)
Permissions (Autorizzazioni)
Supervisione
Inetd
Cron
Backup
Hotplug
PCMCIA
APM
ACPI